



# Electricity Distribution Network Design Under Changing Demand Patterns in Modern Electricity Systems

## **Dissertation**

zur Erlangung der Doktorwürde  
der Wirtschafts- und Verhaltenswissenschaftlichen Fakultät  
der Albert-Ludwigs-Universität Freiburg im Breisgau

vorgelegt von

Alexander Schlüter  
geboren in Backnang

Oktober 2021

Für Hanna

Albert-Ludwigs-Universität Freiburg im Breisgau  
Wirtschafts- und Verhaltenswissenschaftliche Fakultät  
Rempartstr. 16  
79098 Freiburg im Breisgau

Dekan:	Prof. Dr. Matthias Nückles
Erstgutacher:	Prof. Dr. Dirk Neumann
Zweitgutachter:	Prof. Dr. Germain Gaudin

Datum des Promotionsbeschlusses: 29. Oktober 2021



# CONTENTS

ABSTRACT	1
ZUSAMMENFASSUNG	2
<b>CHAPTER I</b>	
MOTIVATION	3
1 Trends in the Electricity Sector . . . . .	4
2 Outline. . . . .	8
<b>CHAPTER II</b>	
BACKGROUND	11
1 Electricity Networks: Generation, Transmission and Distribution . . . . .	11
2 Network Design Problems . . . . .	14
2.1 Distribution Network Design. . . . .	14
2.2 Related Problems in the Electricity Domain . . . . .	17
2.3 Related Problems in Other Domains . . . . .	20

3	Coincidence Factors . . . . .	21
3.1	Idea and Background . . . . .	21
3.2	Coincidence Factors in Network Design . . . . .	24
3.3	The Effect of Load Coincidence on Network Cost and Layout . . . . .	25

## CHAPTER III

### INCLUDING LOAD COINCIDENCE INTO A NETWORK OPTIMIZATION PROBLEM 27

1	The Model . . . . .	29
1.1	Problem Statement . . . . .	29
1.2	Mathematical Formalization . . . . .	30
1.3	Complexity of the CAVLP . . . . .	33
1.4	Alternative Problem Formulation . . . . .	35
1.5	Linearization of the CAVLP . . . . .	36
1.6	Solution Properties . . . . .	40
2	Optimization Methods . . . . .	44
2.1	Generating the Initial Network Layout. . . . .	45
2.2	Improving an Existing Network Layout . . . . .	47
2.3	Optimizing the Capacities. . . . .	53
3	Computational Experiments . . . . .	59
3.1	Experimental Setup . . . . .	59
3.2	Results for High Demand Case . . . . .	61
3.3	Results for High Demand Case . . . . .	65
4	Real-World Experiments . . . . .	69
4.1	Experimental Setup . . . . .	69
4.2	Parameters and Unit Conversions . . . . .	70
4.3	Results. . . . .	72
5	Summary . . . . .	74
6	Additional Information . . . . .	75
6.1	Proof of NP-Hardness for Alternative Problem Formulation . . . . .	75
6.2	Sensitivity Analysis of Capacity Optimization Methods . . . . .	78

## CHAPTER IV

QUANTIFYING THE EFFECT OF LOAD COINCIDENCE ON NETWORK DESIGN	85
1 The Model . . . . .	87
1.1 Solution Properties . . . . .	89
2 Optimization Methods . . . . .	90
2.1 General Optimization Approach . . . . .	90
2.2 Generating an Initial Layout . . . . .	93
2.3 Improving an Existing Network Layout . . . . .	95
2.4 Optimizing the Capacities. . . . .	95
3 Scenarios for Coincidence Factors . . . . .	97
3.1 Background and Assumptions . . . . .	97
3.2 Building the Scenarios . . . . .	97
4 Computational Experiments . . . . .	103
4.1 Experimental Setup . . . . .	103
4.2 Results. . . . .	105
5 Real-World Experiments . . . . .	110
5.1 Experimental Setup . . . . .	110
5.2 Results. . . . .	110
6 Summary . . . . .	112

## CHAPTER V

CONCLUDING REMARKS	113
1 Summary of Findings . . . . .	113
2 Implications . . . . .	115
3 Outlook . . . . .	117

## **CHAPTER VI**

BIBLIOGRAPHY 119

CURRICULUM VITAE 127

LIST OF PUBLICATIONS 129

# LIST OF FIGURES

I-1	Relationship between energy consumption and mean temperatures . . . . .	5
I-2	Projected number of chargers and energy demand for EVs. . . . .	6
I-3	Renewable energy capacity additions per year by technology. . . . .	7
I-4	Outline of this thesis . . . . .	9
II-1	Traditional electricity value chain . . . . .	12
II-2	Today's electricity value chain . . . . .	13
II-3	Relations between the CAVLP and related problems . . . . .	18
II-4	Comparison of various functions for the coincidence factor . . . . .	23
III-1	Data extraction from the geographic information system . . . . .	70
IV-1	Approach to determine initial layout . . . . .	92
IV-2	Comparison of peak demand and peak feed-in per household . . . . .	101
IV-3	Comparison of the coincidence factors for various scenarios . . . . .	102
V-1	Implications for DSOs and regulators . . . . .	115
V-2	Future research topics. . . . .	117



## LIST OF TABLES

III-1	Notation for the CAVLP . . . . .	30
III-2	Parameters for computational experiments . . . . .	60
III-3	Comparison of network cost – low demand. . . . .	62
III-4	Comparison of runtimes – low demand . . . . .	65
III-5	Comparison of infeasible networks – high demand . . . . .	66
III-6	Comparison of network cost – high demand . . . . .	68
III-7	Comparison of runtimes – high demand . . . . .	68
III-8	Comparison of network cost – case study . . . . .	73
III-9	Comparison of runtimes – case study . . . . .	74
III-10	Sensitivity analysis (low demand) – comparison of network cost (MST). . . . .	81
III-11	Sensitivity analysis (low demand) – comparison of network cost (greedy construction) . . . . .	81
III-12	Sensitivity analysis (low demand) – comparison of network cost (random layout) . . . . .	81
III-13	Sensitivity analysis (low demand) – comparison of runtimes (MST) . . . . .	82
III-14	Sensitivity analysis (low demand) – comparison of runtimes (greedy construction) . . . . .	82
III-15	Sensitivity analysis (low demand) – comparison of runtimes (random layout). . . . .	82
III-16	Sensitivity analysis (high demand) – comparison of network cost (MST). . . . .	83
III-17	Sensitivity analysis (high demand) – comparison of network cost (greedy construction) . . . . .	83
III-18	Sensitivity analysis (high demand) – comparison of network cost (random layout) . . . . .	83
III-19	Sensitivity analysis (high demand) – comparison of runtimes (MST) . . . . .	84
III-20	Sensitivity analysis (high demand) – comparison of runtimes (greedy construction) . . . . .	84

III-21	Sensitivity analysis (high demand) – comparison of runtimes (random layout). . . . .	84
IV-1	Notation for the gCAVLP. . . . .	87
IV-2	Parameters for computational experiments . . . . .	103
IV-3	Comparison of network cost, length and branches – main case . . . .	105
IV-4	Comparison of runtimes – main case . . . . .	106
IV-5	Comparison of network cost, length and branches – remaining cases .	107
IV-6	Comparison of runtimes – remaining cases . . . . .	109
IV-7	Comparison of network cost, length and branches – case study. . . .	111

# ABSTRACT

This thesis develops a decision problem that integrates changing demand patterns into the design of electricity distribution networks. Recent trends in the electricity sector change the demand patterns of residential loads. Examples include the coinciding use of air conditioning (AC) during hot summer days, the simultaneous charging of electric vehicles (EVs) over night, but also the coinciding electricity feed-in from photovoltaic (PV) systems. These demand patterns are in contrast to traditional demand patterns that are more stochastic in nature. To model these new demand patterns, a decision problem is developed, namely the *capacitated arborescence with voltage drops and load coincidence problem*. The novelties of the proposed problem are that it considers voltage drops in a more precise manner than any comparable model for network design and that it considers the way that peak loads coincide. Having such a decision problem brings two advantages. First, integrating voltage drops and load coincidence allows for a better network design, i. e., it yields more cost efficient networks while maintaining or improving reliability. Second, it allows to analyze the effect that changing demand patterns have on network design. Therefore, in a first step, problem-specific heuristics are developed to solve the problem. This is needed because the inherent complexity of the problem prohibits exact solutions even for small instances. The heuristics are tested using simulated problem instances and real-world electricity networks. The results point towards considerable cost savings compared to networks designed using conventional planning methods. In a second step, the solution approaches are used to analyze the effect of load coincidence on network cost and layouts. The results indicate that an increase in load coincidence can cause a significant increase of future network cost. In some cases, network cost might even double. The problem can thus provide valuable insights to decision makers such as system operators or regulators.

## Zusammenfassung

Diese Arbeit entwickelt ein Entscheidungsproblem zum Design von elektrischen Verteilnetzen, welches sich verändernde Verbrauchsmuster elektrischer Lasten berücksichtigt. Jüngste Trends im Energiesektor verändern die Verbrauchsmuster von elektrischen Lasten. Beispiele hierfür sind unter anderem die zeitgleiche Nutzung von Klimaanlageanlagen in mehreren Haushalten an heißen Sommertagen, das gleichzeitige Laden von Elektroautos über Nacht, aber auch das gleichzeitige Einspeisen von Strom ins Netz durch Photovoltaikanlagen. Diese Verbrauchsmuster stehen im Gegensatz zu den traditionell eher stochastisch verteilten Verbrauchsprofilen von Haushalten. Um diese neuen Verbrauchsmuster besser zu modellieren, wird ein Entscheidungsproblem entwickelt, welches sich dadurch auszeichnet, dass es zum einen die physikalische Eigenschaft des Spannungsabfalls exakter als vergleichbare Modelle berücksichtigt und dass es zum anderen die Art und Weise, wie Lasten zusammenfallen, mit einbezieht. Ein solches Entscheidungsproblem bringt zwei Vorteile mit sich. Erstens führt es zu besserem Netzdesign, d.h. die entstehenden Stromnetze sind kosteneffizienter, bei gleichbleibender oder verbesserter Ausfallsicherheit. Zweitens erlaubt es, zu analysieren, welchen Effekt sich ändernde Verbrauchsmuster auf das Netzdesign haben. In einem ersten Schritt werden daher Heuristiken zur Lösung des Entscheidungsproblems entwickelt. Dies ist notwendig, denn die Komplexität des Problems macht es unmöglich, exakte Lösungen selbst für kleinste Instanzen zu finden. Die Heuristiken werden dann an simulierten Netzinstanzen, sowie echten Stromnetzen getestet. Die Ergebnisse zeigen klare Kosteneinsparungen im Vergleich zu Netzen, die mit konventionellen Methoden geplant wurden. In einem zweiten Schritt werden die vorher entwickelten Lösungsmethoden verwendet, um den Effekt des Zusammenfallens elektrischer Lasten auf Netzkosten und -aufbau zu analysieren. Im Ergebnis zeigt sich, dass eine stärkere Gleichzeitigkeit des Stromverbrauchs zu einem signifikanten Anstieg der Netzkosten führen kann. In einigen Fällen können sich die Kosten sogar verdoppeln. Das Problem liefert daher wertvolle Einblicke für Entscheidungsträger, wie zum Beispiel Netzbetreiber oder den Regulator.



## MOTIVATION

During a heat wave in August 2020, the Independent System Operator of the U. S. State of California ordered to shut off power to hundreds of thousands of customers for several hours (Burroughs, 2020). These rolling power shutoffs were the first ones in 19 years. The shutoffs were required to avoid an overload of the electricity system, which could have resulted in severe damages or would have caused increased wildfires. All of this was a result of an unexpected increase in electricity demand as Californians tried to cool down their buildings during the heat wave. In an effort to avoid further power shutoffs, the Office of the Governor of California publicly urged people to set their air conditioning to 78°F (25.6°C), avoid major appliance use between 3–10pm, and turn off unnecessary lights (Newsom, 2020).

The U. S. Department of Energy points out that the strain on the electricity system is higher than ever, with parts of the network being more than a century old (U.S. Department of Energy, 2014) and life cycles of network infrastructure typically spanning several decades (IEC, 2015b). Updating this aging infrastructure incurs enormous cost. In the U. S. alone, annual capital investment into distribution networks in 2017 exceeded \$25 billion (U.S. Department of Energy, 2018).

## 1 Trends in the Electricity Sector

Investment cost in electricity infrastructure are expected to further increase due to recent trends in the electricity sector (U.S. Department of Energy, 2020). These trends pose a challenge to decision makers around the globe, ranging from regulators all the way to the private sector. Below, three of these trends are briefly discussed that will largely influence investment decisions in the future. These are (A) direct adaptations to climate change, (B) an increasing share of EVs, and (C) a further decentralized energy generation. All of these trends, of course, are in some way linked to climate change, whether this is in a direct way, as for (A), or in an indirect way, because climate change causes a switch towards a more carbon-neutral energy landscape, as for (B) and (C).

Regarding (A) direct adaptations to climate change, the rolling power shutoffs in California are one example of these adaptations. The shutoffs demonstrate how climate change directly affects electricity systems already today. Deschênes and Greenstone (2011) find that for the U. S., each day with a mean temperature between 80 and 90°F (26.7–32.2°C) increases the monthly energy consumption by 2.0 %, and that each day above 90°F increases energy consumption by 4.4 %. The relationship between temperature and energy consumption is depicted in Figure I-1. The nonlinearity of this relationship implies that small temperature changes can already cause relatively large changes in energy consumption. Davis and Gertler (2015) find a similar dependency between temperature and energy consumption for Mexico. Interestingly, the energy savings from warmer winter days due to less heating are almost negligible compared to the increase in energy resulting from AC usage in the summer. Davis and Gertler (2015) combine their findings with end-of century temperature forecasts and economic forecasts on income growth. In their case study for Mexico, they find that residential electricity consumption might increase by up to 83.1 % by the end of the century due to increasing AC usage. This increase, however, is attributed to both temperature change and a higher disposable income.

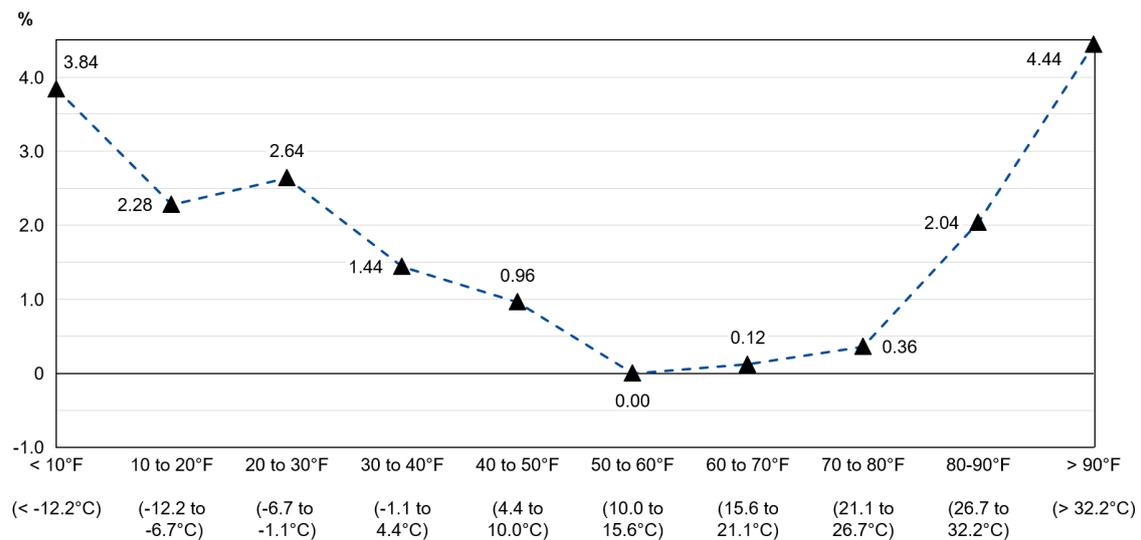


Figure I-1: Estimated relationship between monthly increase in energy consumption and daily mean temperatures in the U. S. Source: Deschênes and Greenstone (2011)

The trends (B) increasing share of EVs and (C) further decentralized energy generation are oftentimes discussed under a common framework because they represent ways of how we try to combat climate change. Following the United Nations Framework Convention on Climate Change (UNFCCC), over recent years, many efforts have been undertaken on a multi-national and national level to combat climate change. In 2015, parties to the UNFCCC signed the Paris agreement (UN, 2015). The agreement aims to keep the increase in the global average temperature below 2°C compared to pre-industrial levels, and to pursue efforts to limit the increase to 1.5°C. It contains specific targets for lowering carbon emissions broken down by country, the so-called nationally determined contributions. These resulted in efforts such as the proposal of a Green New Deal in the U. S. (U.S. House of Representatives, 2019) and the European Green Deal (European Commission, 2019), with the objective of fulfilling these contributions. The European Green Deal states climate neutrality for the EU by 2050 as the target. This target is pursued by cutting pollution, boosting green technologies, creating sustainable industries and transport systems. These overarching themes then translate into national policies, for instance to accelerate the expansion of EVs and renewable energies.

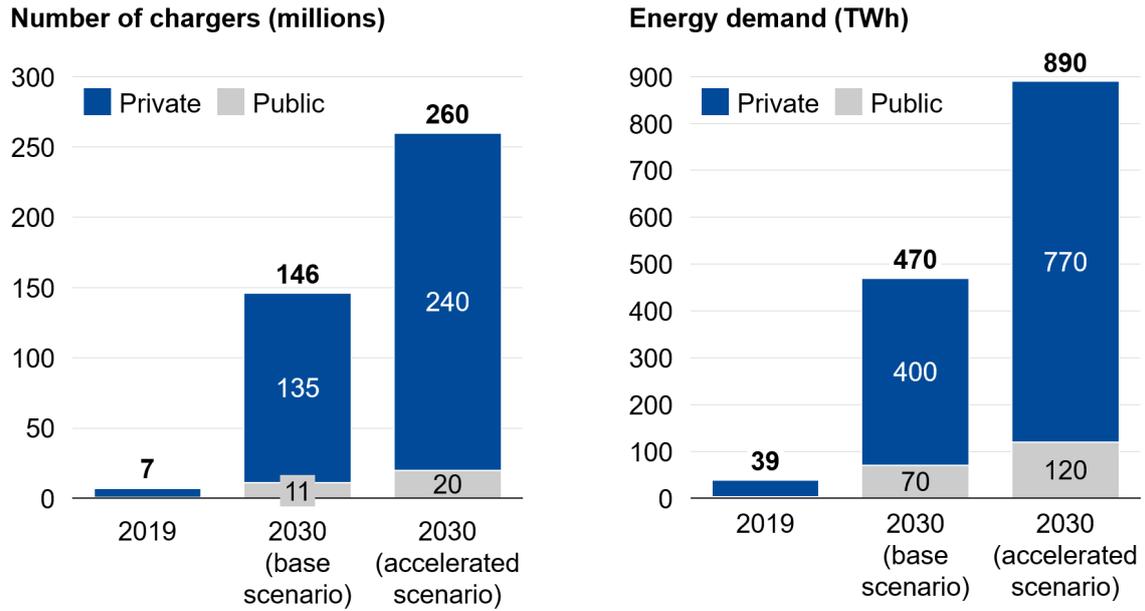


Figure I-2: Left: Projected number of chargers for EVs (in millions). Right: Projected energy demand (in TWh).

Base scenario corresponds to the IEA's "Stated Policies Scenario", accelerated scenario corresponds to the IEA's "Sustainable Development Scenario" which assumes policies in line with limiting the global temperature rise to below 1.7–1.8°C with a 66% probability. Source: IEA (2020)

Regarding (B) increasing share of EVs, the International Energy Agency (IEA) projects the global sales of EVs (including plug-in hybrid vehicles) to increase from 2.1 million vehicles in 2019 to 25–45 million vehicles in 2030 (IEA, 2020). More important for the electricity infrastructure, the number of charging points (both private and public) worldwide is projected to increase from 7 million in 2019 to 140–260 million in 2030. This corresponds to an increase in energy demand for EVs from 39 TWh in 2019 to 470–890 TWh in 2030 (see Figure I-2). This increase in energy demand by a factor of  $\sim 10$ – $20$  over the relatively short time period of 11 years is a challenge for distribution system operators (DSOs) because it can cause overloading of electricity networks. The International Energy Agency states in their Global EV Outlook 2020 that "EV charging can significantly increase and change the timing and magnitude of electricity loads on distribution networks and possibly impact cables, transformers and other components, as well as power quality or reliability" (IEA, 2020). Understanding the implications of EV charging on network infrastructure is therefore a focus topic of Chapters IV and V in this thesis.

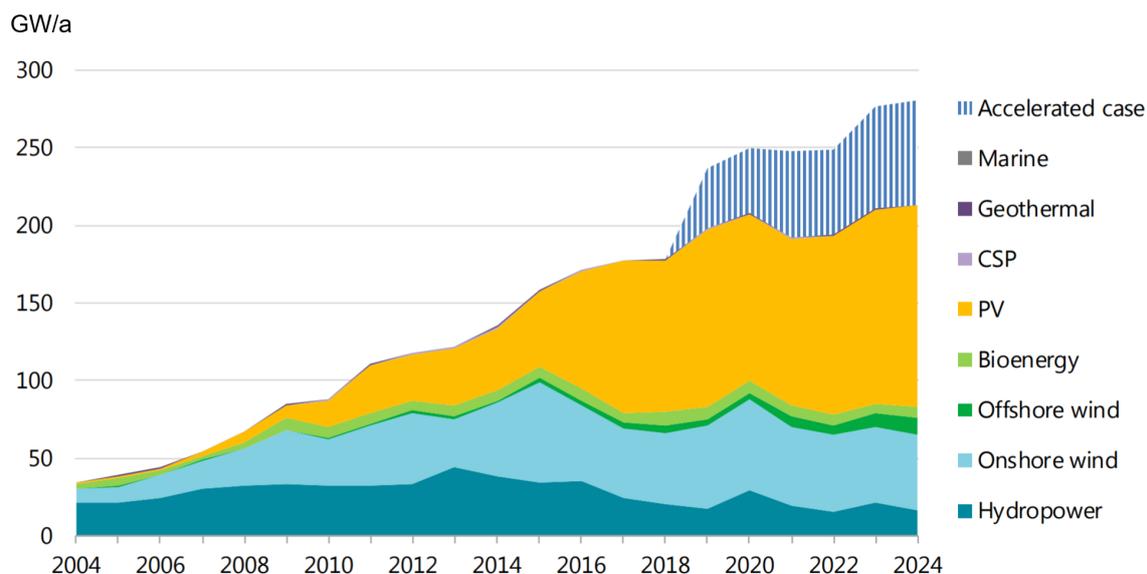


Figure I-3: Renewable energy capacity additions per year by technology. Accelerated case requires policy makers to address regulatory uncertainty, investment risks in developing countries, and system integration of wind and PV in some countries. Source: IEA (2019)

Regarding (C) further decentralized energy generation, this trend is closely linked to an expansion of renewable energy production because renewable energy production is often decentralized. Overall, renewable energy production is projected to increase by 50 % between 2019 and 2024, with photovoltaic (PV) accounting for almost 60 % of all renewable capacity expansion (IEA, 2019). China has been leading the world in renewable energy investments in recent years and accounts by far for the largest PV capacity that is added each year (Mathews and Tan, 2014; UN, 2018). Residential PV systems are growing the fastest, with projections showing a 2.5-fold increase in capacity by 2024 (IEA, 2019). This information is crucial, because residential PV systems are directly connected to the low voltage distribution network, and thus add additional strain on this part of the electricity system. This is the focus of this thesis. To emphasize this point, on the longer term, there is still significant growth potential for residential PV systems. Estimations indicate that even if an aggressive expansion model is applied, by 2024, only 6 % of the technical available rooftop area will be used by distributed PV systems (IEA, 2019).

All of the above mentioned trends do not only change the amount of energy that is produced or generated. They also significantly change demand patterns (i. e., the

way energy is consumed over time). This brings us back to the initial example of AC usage in California, where external factors (in this case: the weather) lead to a high simultaneous electricity use by multiple households. This example shows that it is important to not only look at the effect that these trends have on the average electricity demand or the peak loads of individual households. The way that electricity use of individual households coincides can be crucial because it can cause an overload of the electricity system. As a matter of fact, most of the above mentioned trends have the potential to result in a more coinciding energy use: For instance, charging of EVs is likely to happen in the evening when residents arrive at their home. But not only private charging of EVs is critical in this regard. Gaul et al. (2017) analyzed 450,000 charging sessions at public charging stations in 2014. They found that in 300 hours of the year, all connected cars were charging at full power, thus putting additional strain on the network. Regarding the trend of PV systems, high volumes of PV feed-in can be expected during sunny hours and will also coincide.

As a consequence, new models for electricity network design are needed because existing models use different assumptions on how loads coincide. Often, these models also often involve manual decision-making processes which are inefficient (Gust et al., 2017). Incorporating the way that loads coincide (i. e., load coincidence) into the design of electricity networks brings two advantages. First, more precise design methods yield more cost-efficient networks while maintaining or improving reliability. Second, it allows to analyze the shift in demand patterns that is described above. This means that decision makers (e. g., system operators or regulators) can quantify the effect of increasing load coincidence.

## 2 Outline

The outline of this thesis is shown in Figure I-4, which briefly summarizes content and key results of each chapter. In the next chapter, an introduction to the structure of electricity systems and an overview on network design models is given. In addition

Chapter	Content	Key results
II Background	<ul style="list-style-type: none"> <li>• Providing an overview on related literature on network design and coincidence factors</li> </ul>	<ul style="list-style-type: none"> <li>• A research gap is revealed on works regarding distribution network design that take into account load coincidence and voltage drops</li> </ul>
III Including Load Coincidence into a Network Optimization Problem	<ul style="list-style-type: none"> <li>• Developing a new decision problem for distribution network design</li> <li>• Developing solution methods based on specific problem properties</li> <li>• Conducting computational experiments and a real-world case study to show effectiveness of model and solution methods</li> </ul>	<ul style="list-style-type: none"> <li>• The newly proposed solution methods are computationally tractable</li> <li>• Using the model and the solution methods provides a significant cost benefit when compared to conventional network design</li> </ul>
IV Quantifying the Effect of Load Coincidence on Network Design	<ul style="list-style-type: none"> <li>• Introducing a more generalized version of the decision problem above</li> <li>• Refining the solution methods</li> <li>• Developing scenarios for load coincidence based on trends in the electricity sector</li> <li>• Conducting computational experiments and a real-world case study to analyze the effect of load coincidence on network cost and layout</li> </ul>	<ul style="list-style-type: none"> <li>• Load coincidence can have a severe effect on network cost and layouts</li> <li>• Depending on the scenario and the network size, the network cost can more than double because of load coincidence</li> </ul>
V Concluding Remarks	<ul style="list-style-type: none"> <li>• Deriving implications for various stakeholders</li> <li>• Providing an outlook on potential future research topics</li> </ul>	<ul style="list-style-type: none"> <li>• Changes in load coincidence require action from system operators and regulators</li> <li>• There are open research questions to be explored</li> </ul>

Figure I-4: Outline of this thesis. Shown is the content and the key results of each chapter (excluding the present chapter).

to that, that chapter provides an overview on methods to measure and quantify load coincidence. In Chapter III, these methods are used to develop a decision problem based on integer programming to support the design of electricity distribution networks. Heuristic solution approaches to solve this problem are developed and tested using simulated network instances and 74 real-world electricity distribution networks from a Swiss electricity company. The results show that using the solution approaches can lead to significant cost savings of up to 0.26 million Swiss Francs (CHF) per network. In Chapter IV, the best-performing heuristic from the experiments before is used to determine the effect of different scenarios for load coincidence on network cost and layout. In that chapter, scenarios resembling the trends (A)–(C) are developed. Depending on the scenario, changes in load coincidence can cause the network cost to more than double. This has implications for stakeholders, such as system operators, regulators and others. These are discussed in the final chapter of this thesis. In that chapter, an outlook on future research topics is also provided.



## CHAPTER II

### BACKGROUND

#### **1 Electricity Networks: Generation, Transmission and Distribution**

In the past, the electricity value chain was one-dimensional and allowed for a clear distinction between (1) generation, (2) transmission, and (3) distribution, followed by electricity consumption (cf. Schwab, 2017). (1) Generation was mostly done by large conventional power plants (e.g., coal, nuclear). These power plants were connected to the extra high voltage transmission network. (2) Transmission describes the process of transporting electrical energy over long distances using relatively high voltage. The energy is transported from the point of generation to close to the point of consumption. (3) Distribution describes the process of delivering energy to the end-users (e.g., households) with relatively low voltage. The distribution network is split into a part with mid to high voltage (often referred to as primary distribution network) and a part with low voltage (often referred to as secondary distribution network). Voltage differences in the entire electricity system are bridged by transformers. A schematic representation of this traditional electricity value chain is depicted in Figure II-1. This figure is a simplification of the electricity value chain. For the purpose of this theses, however, this is sufficient, because the present work only aims to understand the effect of general trends on the low voltage distribution network. More precise informa-

tion on the electricity value chain can be found in standard text books such as Schwab (2017). In Figure II-1, electricity generation can be found on the left hand side, while energy consumption is found on the right. Even in this simplified picture, not all generation is connected to the extra high voltage transmission network because some power plants (e. g., wind turbines) are connected to the distribution grid. Also, not all consumers are connected to the low voltage distribution network. Some larger consumers of electrical energy (e. g., factories, airports, large hospitals) are connected to the distribution grid at a higher voltage (Schwab, 2017).

Nowadays, the electricity value chain has slightly changed. A simplified depiction of today's electricity value chain is shown in Figure II-2. There are three main differences when comparing it to the traditional value chain shown in Figure II-1. First, the share of conventional power plants on the overall energy production is lower. Second, the emergence of so-called prosumers leads to energy production at the low

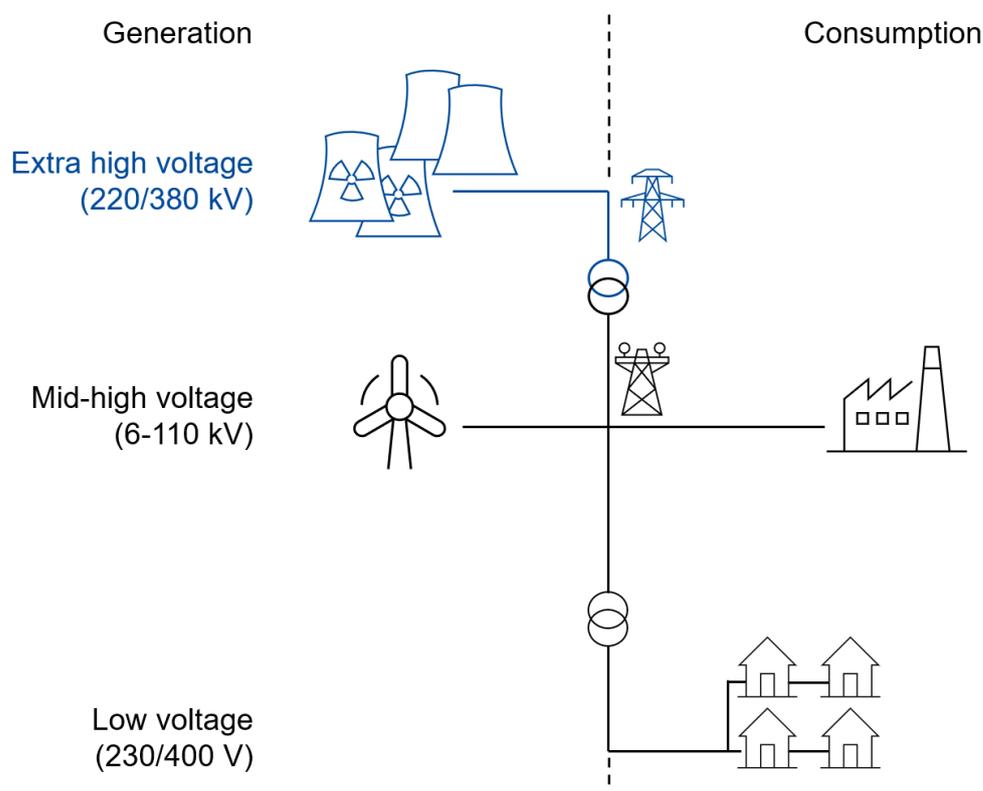


Figure II-1: Traditional electricity value chain (simplified), with generation on the left, consumption on the right. In the past, the electricity value chain was purely one-directional with a clear distinction between generation, transmission, and distribution.

Coloring: Transmission network (and connected components) in blue, distribution network (and connected components) in black.

voltage level. Prosumers are consumers of electrical energy that, at the same time, also produce electrical energy. The most common example of these prosumers are households that own a PV system. Prosumers break the one-directionality of the electricity value chain, because, from time to time, they also feed energy into the network. Third, new technologies, such as EVs and battery storage systems, change demand patterns of consumers. The focus of this thesis lies on the low voltage distribution network, i. e., the network from the last transformer to the end-user (see shading in Figure II-2).

Regarding the layout of the various networks, electricity transmission networks, on the one hand, are typically operated in a meshed layout fulfilling the  $N - 1$ -fail safe criterion, i. e., the network must still be operational, even if one element breaks down.

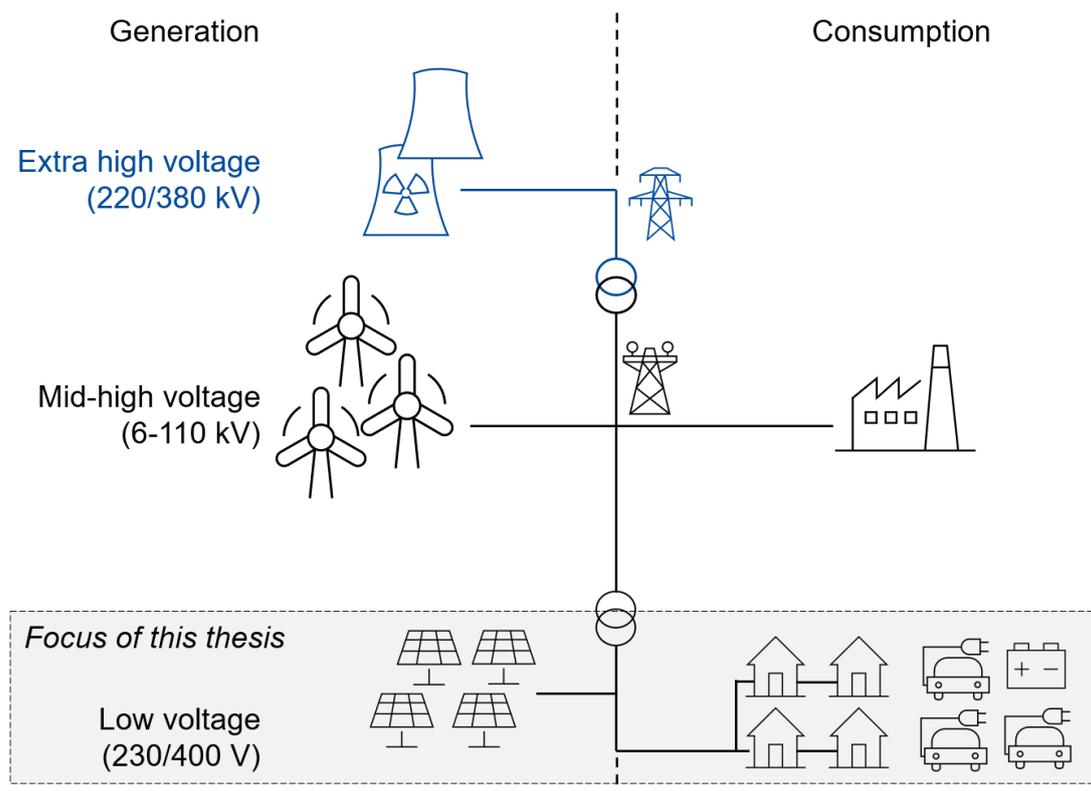


Figure II-2: Today's electricity value chain (simplified), with generation on the left, consumption on the right. Today's electricity value chain exhibits a lower share of conventional power plants (e. g., coal, nuclear). The emergence of prosumers results in energy production at the low voltage level (e. g., by PV systems) and results in a bi-directional power flow. New technologies on the consumer side arise (e. g., EVs and battery storage solutions). The focus of this thesis lies on the low voltage distribution network.

Coloring: Transmission network (and connected components) in blue, distribution network (and connected components) in black.

Thus, every point in the network is served from at least two sides (Schwab, 2017, chap. 10). Distribution networks, on the other hand, are mostly radial networks. This means that they have a tree-like structure. Radial networks are more cost efficient and allow for easier operations and easier maintenance (Schwab, 2017, chap. 11). In this thesis, a problem for distribution network design is presented. Therefore, a radial network layout is one of the requirements for any solution to the proposed problem.

## 2 Network Design Problems

### 2.1 Distribution Network Design

Distribution networks deliver electricity to households by connecting them to the transmission part of the electricity system. The design of distribution networks is a non-trivial task. The networks need to satisfy the following three constraints (Weedy, 2012). (i) The capacity of each grid line must meet the variable electricity demand and thus be sufficiently large. (ii) The layout of the network must be radial. (iii) Due to the electric resistance of the grid lines, voltage drops (i. e., a reduction in the electric voltage along the grid lines) occur, whose magnitude must remain below a certain threshold (cf. CENELEC, 2010; IEC, 2015a). Solving the problem with constraints (i)–(iii) has only been achieved by restricting the number of possible layouts (Georgilakis and Hatziaargyriou, 2015).

Modeling the constraints (i) and (ii) without voltage drops yields a capacitated shortest spanning arborescence problem (Chandy and Lo, 1973) with multiple available line capacities. This problem was solved in prior research (e. g., Brimberg et al., 2003; Singh et al., 2009). Accounting for voltage drops in distribution network design, however, is of critical relevance. The reason is that voltage drops beyond a certain threshold can damage the network and connected electrical devices. Modeling of voltage drops is challenging as it involves multiple parameters (Weedy, 2012). More precisely, voltage drops depend on the flow of electric current  $F$ , the line length  $l$ ,

and the cross section (i. e., capacity)  $a$  of the grid line. In practice, voltage drops beyond the critical threshold are commonly resolved manually during the design process (Gust et al., 2017).

This work extends research on distribution network design by presenting a decision model that includes all aforementioned constraints, namely (i) line sizing, (ii) radial layout, and (iii) voltage drops, whilst putting no additional restrictions on the network layout. A key novelty of this model is, that when considering constraint (i) line sizing, the model takes into account load coincidence. The objective of this model is to identify the cost-optimal network that connects pre-defined locations of electricity demand to a source. The output determines how to connect these locations with grid lines and which capacity to choose for these connections.

Several works have addressed the above mentioned constraints (i)–(iii) (de Boeck and Fortz, 2018; Qi et al., 2015; Rossi et al., 2012; Singh and Mason, 2004; Singh et al., 2009). However, none of these works model the combination of constraints (i)–(iii) in an exact manner—whereas we rigorously model all constraints, particularly voltage drops and load coincidence. Below, the modeling of each constraint is reviewed and the differences to this work are outlined.

(i) *Line sizing* guarantees that the capacity of each grid line is sufficient to supply the connected loads. In practice, electricity companies have several different line types at their disposal and can choose among them. Each line type has a different cross-sectional area. Some works, however, ignore the existence of a discrete set of line types (e. g., Nahman and Peric, 2008; Rossi et al., 2012), while in other works multiple line types with different capacities are included (e. g., Singh et al., 2009; Qi et al., 2015). When choosing the line sizes, network designers often only consider individual peak loads of households. Since these peak loads do not coincide (Dickert and Schegner, 2010), this worst case scenario can result in an over-sizing of the network. Taking into account this non-coincidence of peak loads when aggregating loads can result in significant cost savings. This thesis considers the degree of load

coincidence by using the coincidence factor (see, e. g., Beaty and Fink, 2013; Gonen, 2015). A detailed description of the coincidence factor, including an overview on network problems that consider load coincidence is discussed later (Section 3). Unless stated otherwise, conventional distribution network design models do not take load coincidence into account.

(ii) A *radial layout* is required so that the flows of energy from the source to each load follow a unique path (cf. Singh et al., 2009). Owing to this, the design of electricity distribution networks is related to mixed-integer arborescence problems, in particular, the capacitated shortest spanning arborescence rooted at  $r$  ( $CSSA_r$ ) problem (Chandy and Lo, 1973; Papadimitriou, 1978; Toth and Vigo, 1995). The objective of the  $CSSA_r$  problem is to find a directed rooted tree (i. e., an arborescence) with minimum cost, such that the demands in each branch do not exceed a certain value.

(iii) *Voltage drops* are a core element of the problem presented in this thesis. Despite their importance for network design practice, voltage drops have not yet been rigorously modeled in greenfield network design. Along the lines of any electricity system, the voltage steadily drops due to the electrical resistance of the lines, leading to a lower voltage at the end-point of the system. Voltage drops beyond a certain threshold can damage the electric system and the connected electrical devices and, hence, must be avoided when designing a network. This has a substantial effect on determining which network layouts are physically feasible (Adams and Laughton, 1974). There are three approaches to including voltage drops in network design: First, there is the widely used industry practice of manually considering voltage drops during the design process (Gust et al., 2017). Second, there is the approach of using alternative metrics as a proxy for voltage drops. De Boeck and Fortz (2018) and Rossi et al. (2012) use a technique called “hop-constraining”, which prohibits any vertex of the network from being any more than a given number of steps (“hops”) away from the source. An overview of the hop-constrained minimum spanning tree problem can be found in Dahl et al. (2006). This approach is also used by Requejo and Santos (2009) and Gouveia and Magnanti (2003), who propose a heuristic approach to solving the

hop-constrained minimum spanning tree problem. In hop-constraining, the voltage drop is simplified, that is, both the actual length of the lines and the electric current (i. e., the flow) are omitted. Furthermore, hop-constraining is also limited to problems with a single line capacity. It is therefore not applicable for problems where more than one line type is available, and where line lengths and flows vary substantially across the network. Third, there is the approach of using an explicit model for voltage drops and including a minimum allowable voltage as a direct constraint. As is shown in this work, this approach implies significant computational complexity in the model. Here this thesis contributes a tractable solution approach using heuristics.

There are some approaches that take into account constraints (i)–(iii) for network design, but at the same time restrict the network layouts. These approaches are predominantly covered in electricity systems engineering and are reviewed in Georgilakis and Hatziaargyriou (2015). All of these publications put restrictions on network layouts, e. g., by restricting connections to vertices that are close in terms of Euclidean distance (Boulaxis and Papadopoulos, 2002; Gan et al., 2011), close in terms of an existing street layout (Kong et al., 2009; Navarro and Rudnick, 2009), or close in terms of pre-existing network infrastructure (Cossi et al., 2005; Falaghi et al., 2011). Therefore, the aforementioned approaches disregard many potentially cost-effective layouts where line connections are long. Later, heuristics are presented that modify the network layout by introducing longer connections and show that these connections can reduce network cost.

## **2.2 Related Problems in the Electricity Domain**

Distribution network design is closely related to distribution network reconfiguration, power flow modeling, and transmission network design. Figure II-3 summarizes the similarities and differences between the problems in these fields and the problem developed in this thesis, namely the capacitated arborescence with voltage drops and load coincidence problem (CAVLP). The relations are summarized below.

	Optimize for investment cost	Layout optimization	Capacity optimization	Unrestricted layouts	Radiality constraint	Explicit model for voltage drops	Load coincidence
Distribution network reconfiguration		✓			✓	⋯	
Power flow modeling						✓	⋯
Transmission network design	✓	✓	⋯			⋯	
CAVLP	✓	✓	✓	✓	✓	✓	✓

Figure II-3: Relations between the CAVLP and related problems. Similarities are indicated by a tick mark. Dashed tick marks indicate that the models sometimes or partly fulfill the respective property. Similarities can be found in the objective (i. e., whether the problem optimizes for investment cost) or the constraints.

Distribution network reconfiguration concerns the modification of existing networks with the objective of improving operations (e. g., Avella et al., 2005; Glamocanin, 1990; Jabr et al., 2012; Parada et al., 2010). This stream of research is closely related because the networks need to satisfy the same constraints (i)–(iii). There are, however, significant differences. First, network reconfiguration problems optimize for minimum operational cost by minimizing power losses. In contrast, this thesis focuses on minimizing investment cost. Both objectives yield different solutions (e. g., a grid line with a large cross section comes with high material cost but results in low operating cost due to low power losses). Second, this stream of research also restricts the number of potential layouts as only a limited amount of edges can be activated or deactivated. For instance, the largest network instances considered by Avella et al. (2005) are comparable to ours in the number of vertices. However, the authors consider only around 150 alternative edges (i. e., switches). Other approaches (e. g., Freitas et al., 2016) allow even fewer modifications to the layout. In the CAVLP, comparable instances exhibit over 10,000 possible alternative edges. Third, the capacity (i. e., line type) for any given connection is usually fixed in network reconfiguration problems. The CAVLP considers multiple possible line types. Adapting the reconfiguration approaches for network design purposes is thus non-trivial. Nevertheless, it is possible

to re-purpose models from the reconfiguration literature by introducing such a large amount of switches. This approach is later used as a benchmark in evaluating the performance of the developed solution methods.

The literature stream on power flow modeling considers the same physical principles as the problem developed in this thesis. Yet, instead of designing a network, it is concerned with finding solutions to the power flow equations for a given network layout (Glover et al., 2012). In other words, the network is assumed a fixed parameter, the power injections at the sources (i. e., the transformers) are given, and the problem is to find the voltages at each node. Most notably, the optimal power flow (OPF) problem aims at finding the most efficient way to operate electricity networks (i. e., minimizing generation costs) under steady-state conditions (Zohrizadeh et al., 2020). Owing to this, this stream of literature focuses on numerical algorithms and approximations, for instance, the LinDistFlow model and recent alterations thereof (Molzahn et al., 2017; Schweitzer et al., 2020). The modeling of current flows and voltage drops presented in this thesis closely follows the LinDistFlow model. This also means that a DC approximation for the currents is used instead of using AC. This approximation is widely used in the literature (see e. g., Avella et al., 2005).

The literature on transmission network design aims at optimizing the long-range electricity delivery and the underlying high voltage networks (e. g., Fisher et al., 2008; Kocuk et al., 2016; Khodaei et al., 2010; Villumsen and Philpott, 2012), often in combination with power plant planning (e. g., Jenabi et al., 2015; Märkle-Huß et al., 2019; Pineda and Morales, 2016). While the constraints (i) and (iii) in these problems are similar to those in the CAVLP, these problems disregard the radiality constraint (ii), as transmission networks usually operate in a meshed layout fulfilling an  $(N - 1)$ -fail-safe criterion. Additionally, layouts are fairly restricted as layout changes are achieved by switching operations of a very limited number of grid lines. In sum, the transmission network design problems are more distant to the CAVLP than the distribution network reconfiguration problems, because of the absence of radiality constraints and the smaller number of potential layouts. Therefore, a model from the distribution net-

work reconfiguration literature is used as a benchmark in evaluating the performance of our solution methods instead of a model from the transmission literature.

### 2.3 Related Problems in Other Domains

Similar physical properties to voltage drops also occur in network problems from other domains. For instance, in pipeline systems, drops in pressure depend on pipe diameter, flow, and pipe length and, as such, must be constrained in distribution networks for oil (Brimberg et al., 2003) or gas (André et al., 2013). In transportation networks, traversal time or reliability constraints can have similar mathematical forms to voltage drops (Balakrishnan et al., 2017). Also, the principle of coinciding demand/supply exists as well, for instance in studies on water supply or flood analysis (Chen et al., 2012; Yan and Chen, 2013).

Despite these similarities, the above problems have inherent differences to the CAVLP. For instance, transportation networks are usually not subject to radial layouts. More importantly, the constraints related to the drop (e. g., pressure drops) are oftentimes simply omitted due to the computational complexity (e. g., Brimberg et al., 2003). Only very few prior works explicitly model voltage drops: André et al. (2013), for instance, incorporate pressure drops in a gas distribution network. Their problem also involves radial layouts, yet their line sizing is based on continuous capacities and does not take into account the non-coincidence of peak demands. The problem presented in this thesis involves a discrete set of capacities and does take into account varying degrees of demand coincidence. Both characteristics introduce additional complexity and alter efficient network layouts, as shown later.

In summary, prior work has not combined all constraints (i)-(iii) without imposing additional constraints that can lead to sub-optimal network designs. However, the above examples demonstrate the widespread applicability of the CAVLP and the proposed solution methods for network design.

Despite the above mentioned shortcomings, the work by (Brimberg et al., 2003) is highly relevant for this thesis. Their problem concerns the design of a radial distribution network for oil pipelines that connect oil wells to one single harbor. In a case study, the authors design a distribution network for a real-world oil field in South Gabon. In terms of size (i. e., the number of vertices), this network is comparable to our problem instances. The problem is solved heuristically by two metaheuristics, namely Tabu Search and Variable Neighborhood Search (VNS). This approach is adapted here, as this thesis also uses these two metaheuristics alongside other solution approaches for solving the CAVLP (see Section 2).

Finally, while all of the problems above ignore the existence of load coincidence, there are situations, where the non-coinciding nature of demand is core to the problem. This becomes most evident when looking at problems on precipitation and drainage. For instance, Chen et al. (2012) estimate the flood risk from precipitation in the Yangtze River and the Colorado River. Rainfalls that are subject to a certain stochasticity at various branches of the river can lead to flooding in the main river. Therefore, the problem exhibits an arborescence structure and also includes a form of load coincidence. Yan and Chen (2013) investigate irrigation and analyze if there is enough precipitation to guarantee reliable water supply in a large water transfer project in China. This literature stream can benefit from both the model presented here and the corresponding solution approaches.

## **3 Coincidence Factors**

### **3.1 Idea and Background**

Electricity networks do not have to be built to withstand the theoretically highest-possible peak load. This is because the electricity demands of multiple loads do not all peak at the same time. Thus the expected peak demand of a group of loads does

not equal the sum of the individual peak demands (Dickert and Schegner, 2010). This phenomenon is called load coincidence (or load diversity).

Therefore, when building a network, instead of considering the theoretically highest-possible peak load, the goal is to meet a scenario, where the grid lines have to withstand the expected peak flows. In a radial distribution network, this expected peak flow equals the coincident peak demand of all connected loads. This aggregation of loads is typically done by considering a discount factor, namely the coincidence factor. This discount factor describes how the peak demand per load decreases as the number of loads increases. It is defined as the peak demand of a group of loads divided by the sum of the peaks of the individual loads. The reciprocal of the coincidence factor is called diversity factor. The terminology, however, is not always consistent and the two terms are often used interchangeably. A review of various mathematical functions for the coincidence factor can be found in Dickert and Schegner (2010). In Figure II-4, we show three different functions for illustration. The first two functions follow a formula derived in Dickert and Schegner (2010) and are given by

$$\gamma(N) = \gamma_{\text{lim}} + (1 - \gamma_{\text{lim}}) N^{-1/2}. \quad (\text{II-1})$$

The two functions shown in Figure II-4 have different values for  $\gamma_{\text{lim}}$  and thus converge towards  $\gamma_{\text{lim}} = 0.1$  and  $\gamma_{\text{lim}} = 0.3$ , respectively. The other function shown in the figure is empirically determined in Nickel and Braunstein (1981) in an effort to match several curves published in the U. S. This function is given by

$$\gamma(N) = 0.5 \cdot \left( 1 + \frac{5}{2N + 3} \right). \quad (\text{II-2})$$

Load coincidence and the corresponding coincidence factors have been subject to intensive empirical research because they depend on various factors, such as geographics, degree of electrification, etc. (Boait et al., 2015; Herman and Kritzing, 1993; Konstantelos et al., 2014; Richardson et al., 2010; Widén and Wäckelgård,

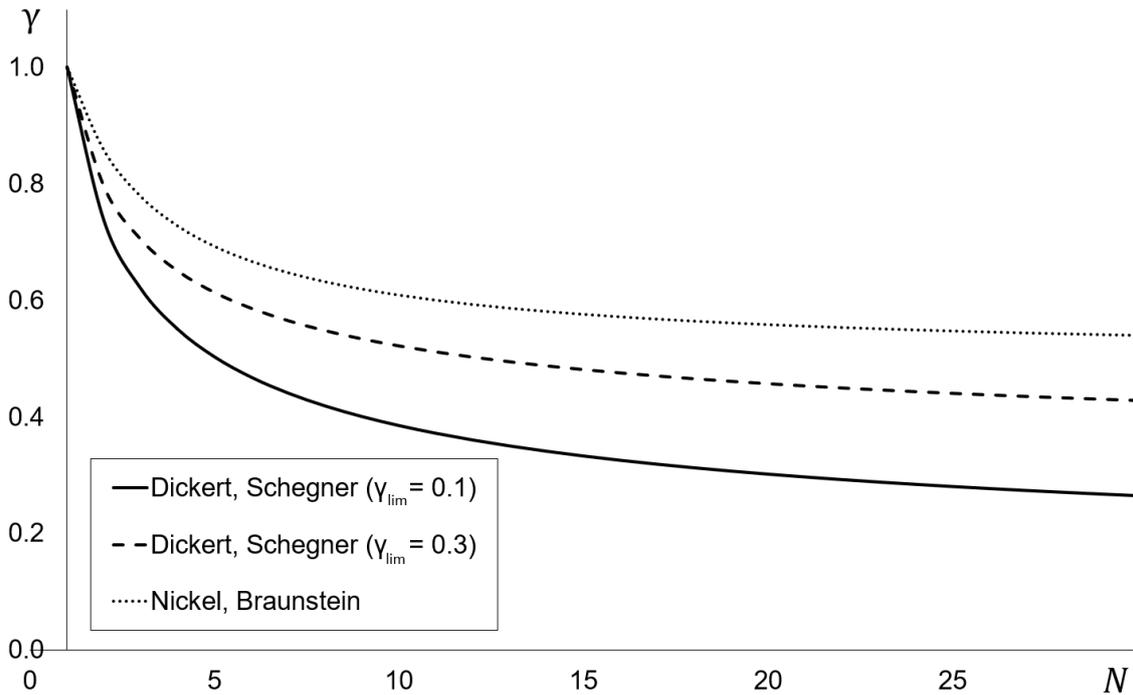


Figure II-4: Comparison of various functions for the coincidence factor  $\gamma(N)$  as a function of the loads  $N$ . Shown are two functions as derived in Dickert and Schegner (2010) (one converging towards  $\gamma_{lim} = 0.1$ , another converging towards  $\gamma_{lim} = 0.3$ ) and one function as derived in Nickel and Braunstein (1981).

2010). For instance, Herman and Kritzing (1993) analyze load data from South Africa to model the coincidence of residential demands in a descriptive manner. Their approach has widely been picked up by researchers. More recent publications use prescriptive bottom-up models of simulated appliance usage to derive demand patterns of individual households and groups of households (Richardson et al., 2010; Widén and Wäckelgård, 2010). The spread of smart meters in recent years has made it possible to gather real-time data from thousands of households within the same network (Konstantelos et al., 2014). This allows for a more precise determination of coincidence factors.

All functions of the coincidence factor (whether derived mathematically or determined empirically) have the following properties in common. For one household ( $N = 1$ ), the value for the coincidence factor is always 1, i. e., no discount is applied. Further, the functions are monotonically decreasing in  $N$ , convex, and approach a threshold  $\gamma_{lim}$  as networks become larger.

A key complication when considering coincidence factors in the context of network design is that flow conservation does no longer apply. Specifically, because the discount is higher, the more loads are located downstream, the peak flow into a vertex will be less than the outgoing flows and the demand of this vertex combined. Therefore, equations concerning the power flow must be adjusted.

### **3.2 Coincidence Factors in Network Design**

Some researchers utilize coincidence factors to improve the operation of distribution networks (e. g., Resch et al., 2017; Boait et al., 2015). However, there is a dearth of publications incorporating load coincidence into the design of distribution networks. Below, related publications are discussed. It is also pointed out, where the main differences to the work presented in this thesis lie.

Domingo et al. (2011) use coincidence factors in the design of distribution networks. They consider a large network with three layers, namely a high voltage layer, a medium voltage layer and a low voltage layer. Coincidence factors, however, are only applied at connections between these layers, i. e., for the transformers. This means that, for instance, each household in the low voltage network gets discounted by the same amount. Within the low voltage network, coincidence factors are simply ignored. Similarly, Parshall et al. (2009) use coincidence factors to estimate the demands of entire LV distribution networks as a whole to help design the upstream parts of the electricity system. Kaur and Sharma (2008) and Sauhats et al. (2016) present models that take into account load coincidence for network design. However, they only choose conductors (i. e., line cross sections, conductor types, etc.) for a fixed network layout.

So far, there is no publication on designing a electricity distribution network that takes into account coincidence factors on the grid line level. The model presented in this thesis bridges this gap. For every vertex in the network, the model considers the load coincidence of the demands downstream and applies the correct discount.

### **3.3 The Effect of Load Coincidence on Network Cost and Layout**

In this thesis it is also analyzed, how various functions of the coincidence factor affect network cost and network layout. There is only limited research on this topic. This is, in part, because of the lack of suitable models, as pointed out above. Gwisdorf et al. (2010) investigate the effect of coinciding demand side management (DSM). They show how an uncoordinated DSM can lead to higher investment cost as it requires network reinforcement measures. Contrasting to this, Vallés et al. (2016) argue that a coordinated DSM can also help to reduce peak loads and defer investments. Another technology, namely EVs, is investigated by Richardson et al. (2010). They show that a higher EV penetration increases peak loads in a distribution network. Similar observations have been made by Wieland et al. (2015). In an empirical study, Gaul et al. (2017) analyzed 450,000 charging sessions at public EV charging stations in 2014. From their observations, the authors conclude that a coincidence factor of 1.0 must be assumed when describing the underlying distribution network. Later, this observation is used in the development of various scenarios for load coincidence.



## CHAPTER III

# INCLUDING LOAD COINCIDENCE INTO A NETWORK OPTIMIZATION PROBLEM

In this chapter, a decision problem based on integer programming is presented. The problem supports the design of electricity distribution networks. To this end, the capacitated arborescence with voltage drops and load coincidence problem (CAVLP) is proposed. The problem models voltage drops as a key physical constraint of the problem while keeping the network layouts unrestricted. At the same time, it considers load coincidence, i. e., non-coinciding peak loads. The model further considers other constraints, including line capacities and radiality of the layout. The inherent complexity of the CAVLP prohibits exact solutions even for small problem instances. Therefore, a set of problem-specific heuristics is created and evaluated based on simulated network instances. Additionally, the approach is tested on 74 electricity distribution networks from a Swiss electricity company. The experiments demonstrate that the proposed solution approach is computationally tractable. Furthermore, when compared to conventional design practice, they point towards considerable cost savings over the actual network layouts of more than 39%. This corresponds to cost savings of up to CHF 0.26 million per network.

In summary, the contributions of the work presented in this chapter are five-fold:

1. An approach to distribution network design is presented which accounts for voltage drops at the design stage. This model also considers load coincidence. The corre-

sponding decision problem is the CAVLP. Voltage drops and load coincidence are included in the model in a way that does not restrict network layouts beyond radiality. Thus, the CAVLP overcomes shortcomings of existing problems for distribution network design.

2. It is proven that the CAVLP is NP-hard. Furthermore, it is shown that the problem contains complex nonlinearities. Linearizations for the problem are provided. It is shown that the complexity of the problem makes it intractable to use exact solvers (e.g., through branch-and-bound), even for smallest instances with 10 vertices or fewer. In addition to that, an alternative problem formulation based on arborescences is presented. This alternative formulation also yields an NP-hard problem. Although this formulation avoids complex nonlinearities, it is inferior from a computational standpoint because the number of constraints grow exponentially in the number of loads.
3. Given the aforementioned complexity, the CAVLP is solved by developing heuristics. The heuristics are designed in such a way that they leverage the unique physical properties of the problem.
4. The heuristics are evaluated across different simulated problem instances to analyze their solution quality. Also, upper and lower bounds for the exact solution are provided using simplified problem instances for comparison. This confirms the effectiveness of the heuristics across various problems instances.
5. The effectiveness of the solution approach is shown based on 74 real-world distribution networks from a Swiss electricity company. Thereby, the existing networks that followed a conventional design from practitioners are compared against the ones from the proposed solution approach. The experiments result in improvements of more than 39%. This corresponds to cost savings in the order of up to CHF 0.26 million per network. Altogether, these findings highlight the practical value-add of the work presented.

# 1 The Model

## 1.1 Problem Statement

Electricity distribution network design corresponds to the decision problem of connecting a given set of demand locations (e. g., households) with a single source location (i. e., the transformer to the superordinate network). Between the locations, grid lines of different types can be built. Each line type has a specified capacity (i. e., its cross section). The objective for the decision maker is to minimize investment costs consisting of construction and material costs. The problem is subject to the following constraints.

- (i) *Line sizing.* The capacity of a grid line must be large enough to support the electric current (i. e., the flow). When choosing the capacity, we take into account that the peak demands of individual loads are not coinciding. For every line in the network, we discount the flow by the line-specific coincidence factor.
- (ii) *Radial layout.* Networks layouts must be radial, so that all energy flows from the source to each load follow a unique path.
- (iii) *Voltage drops.* The flow of electric current through a line causes a voltage drop. The voltage drop accumulates over consecutive lines. At any point in the network, it must remain below a threshold prescribed by industry norms.

## 1.2 Mathematical Formalization

The previous decision problem is formulated as the capacitated arborescence with voltage drops and load coincidence problem. An overview of the notation is provided in Table III-1. The appropriate unit conversions and material constants for real-world settings can be found in Section 4.2.

Symbol	Description	Unit/range
$G$	Directed multigraph	$G = (V, E)$
$V$	Set of all vertices	
$N$	Number of vertices	$N =  V $
$i, j$	Indices of vertices	$i, j = 0, \dots, N - 1$
$D_i$	Demand of vertex $i$	$D_0 = 0, D_{i \neq 0} = D > 0.$
$E$	Set of all directed edges	
$k$	Index of line type	$k = 1, \dots,  A $
$(i, j)^k$	Directed edge from $i$ to $j$ ; $k$ denoting its type	$(i, j)^k \in E$
$A$	Set of edge capacities depending on line type $k$ (in ascending order)	
$a_{ij}^k$	Edge capacity	$a_{ij}^k > 0, a_{ij}^k \in A$
$N_j$	Set of all vertices that can be reached from $j$	
$\Gamma$	Graph representing one solution of the problem	$\Gamma = (V, E'), \text{ with } E' = \{(i, j)^k \in E \mid x_{ij}^k = 1\}$
$\Gamma_j$	Subgraph of $\Gamma$ including $j$ and all edges and vertices reachable from $j$	
$ I_j $	Number of vertices in $I_j$	
$\gamma( I_j )$	Coincidence factor (discount factor depending on number of vertices)	$0 < \gamma( I_j ) \leq 1$
$\bar{D}_i$	(Un-discounted) sum of all demands in $I_j$	
$d(i)$	Depth of vertex $i$ , i. e., number of hops to reach $i$ from the source in $\Gamma$	
$F_{ij}$	Flow through edge $(i, j)^k$	$F_{ij} > 0$
$l_{ij}$	Length of edge $(i, j)^k$	$l_{ij} \in \mathbb{R}^+$
$P$	Set of all paths from source vertex 0 to any leaf vertex	Set of edge sequences
$p$	Specific path from source vertex 0 to a leaf vertex	Edge sequence, $p \in P$
$c_c$	Construction costs	Monetary unit per distance
$c_m$	Material costs	Monetary unit per distance per capacity unit
$U_i$	Voltage at vertex $i$	$U_i > 0$
$U$	Voltage at transformer	$U > 0, U_0 = U$
$U_{\text{crit}}$	Critical voltage level	$U_{\text{crit}} > 0$
$Q$	Threshold value for voltage drop	$Q = U - U_{\text{crit}}$
$x_{ij}^k$	Decision variable for edge from vertex $i$ to $j$ with capacity $a_{ij}^k$	$x_{ij}^k \in \{0, 1\}$

Table III-1: Notation for the CAVLP.

Let  $G = (V, E)$  denote a complete directed multigraph without loops. The set of vertices  $V = \{0, \dots, |N - 1|\}$  represents locations. Each vertex  $i \in V$  has a given demand  $D_i$ . The source location (i. e., the transformer) is defined as vertex 0 and we further set  $D_0 \stackrel{\text{def}}{=} 0$ . The set of edges  $E$  contains all potential grid lines. Each edge  $(i, j)^k \in E$  from vertex  $i$  to vertex  $j$  has a discrete capacity  $a_{ij}^k \in A$ . The index  $k$  indicates the line type. Furthermore, each edge  $(i, j)^k$  has a given length  $l_{ij}$  that is independent of its type (i. e., the same for all  $k$ ). The decision variable  $x_{ij}^k \in \{0, 1\}$  indicates whether an edge of type  $k$  from vertex  $i$  to vertex  $j$  should be built. The subgraph representing one solution of the problem is denoted  $\Gamma(x) = (V, E')$  with  $E' = \{(i, j)^k \in E \mid x_{ij}^k = 1\}$ . Furthermore, let  $\Gamma_j$  denote the subgraph of

$\Gamma$ , encompassing a certain vertex  $j$  and all vertices and edges that can be reached from  $j$  in the direction of the flow. The number of vertices in  $\Gamma_j$  is denoted  $|\Gamma_j|$ . The CAVLP is then given by

$$\min \sum_{(i,j)^k \in E} x_{ij}^k [l_{ij}c_c + l_{ij}c_m a_{ij}^k] \quad (\text{III-1})$$

$$\text{s. t.} \quad \sum_{k \in \{1, \dots, |A|\}} x_{ij}^k a_{ij}^k \geq F_{ij}, \quad \forall i, j \in V, \quad (\text{III-2})$$

$$\begin{aligned} \sum_j F_{ji} - \sum_j F_{ij} &= \gamma (|\Gamma_i|) D_i - \\ &\sum_j \sum_k x_{ij}^k ([\gamma (|\Gamma_j|) - \gamma (|\Gamma_i|)] \bar{D}_j), \quad \forall i \in V \setminus \{0\}, \end{aligned} \quad (\text{III-3a})$$

$$|\Gamma_i| = 1 + \sum_j \sum_k x_{ij}^k |\Gamma_j|, \quad \forall i \in V, \quad (\text{III-3b})$$

$$\bar{D}_i = D_i + \sum_j \sum_k x_{ij}^k \bar{D}_j, \quad \forall i \in V, \quad (\text{III-3c})$$

$$|\Gamma_0| = N - 1, \quad (\text{III-3d})$$

$$\bar{D}_0 = \sum_j D_j, \quad (\text{III-3e})$$

$$\sum_i \sum_k x_{ij}^k = 1, \quad \forall j \in V \setminus \{0\}, \quad (\text{III-3f})$$

$$\sum_k x_{ij}^k \frac{a_{ij}^k}{l_{ij}} (U_i - U_j) = F_{ij}, \quad \forall i, j \in V, \quad (\text{III-4a})$$

$$U_i \geq U_{\text{crit}}, \quad \forall i \in V, \quad (\text{III-4b})$$

$$U_0 = U. \quad (\text{III-4c})$$

The objective in Equation (III-1) is to minimize investment costs. If an edge with capacity  $a_{ij}^k$  from vertex  $i$  to vertex  $j$  is built, construction costs of  $l_{ij}c_c$  are incurred (depending only on the length of the edge) and material costs of  $l_{ij}c_m a_{ij}^k$  are incurred (depending on the length and the capacity).

The line sizing constraint in Equation (III-2) requires the capacity  $a_{ij}^k$  of an edge  $(i, j)^k$  to be sufficiently large to support the peak flow  $F_{ij}$ .

Equations (III-3a) to (III-3f) define the flows and ensure the radial layout (including connectivity) of the network. The peak flows  $F_{ij}$  are defined recursively in Equation (III-3a). In the special case of a uniform coincidence factor of  $\gamma \equiv 1$ , Equation (III-3a) simplifies to  $\sum_j F_{ji} - \sum_j F_{ij} = D_i$  and Equations (III-3b) and (III-3c) are not required. In this special case, all flows into a vertex minus all flows out of this vertex equal the demand of the vertex. Except for this special case, the peak flows in the CAVLP are not conserved because of the non-coincidence of peak loads. To take this non-coincidence into account, the correction term in the second line of Equation (III-3a) is needed, which is explained as follows. The correction term considers the direct neighbors of vertex  $i$ , i. e., all vertices  $j$  with  $\sum_k x_{ij}^k = 1$ . Each neighbor  $j$  connects a subgraph  $\Gamma_j$  to vertex  $i$ . The number of vertices in a subgraph  $|\Gamma_j|$  determines the magnitude of the coincidence factor  $\gamma(|\Gamma_j|)$  and thus the flows going out of vertex  $i$ . The correction term determines the discount of outgoing flows relative to incoming flows, which is given by the difference in the coincidence factors  $\gamma(|\Gamma_j|)$  and  $\gamma(|\Gamma_i|)$ . Equations (III-3b) and (III-3c) define  $|\Gamma_i|$  and  $\bar{D}_i$  recursively: The number of vertices inside a subgraph  $|\Gamma_i|$  always equals one plus the sum of all subgraphs downstream.  $\bar{D}_i$  denotes the (un-discounted) sum of demands downstream a certain vertex  $i$ . This is equal to  $D_i$  plus the sum of all demands downstream. Equations (III-3d) and (III-3e) define vertex 0 as the source by setting the values for  $|\Gamma_0|$  and  $\bar{D}_0$  to the overall number of loads and the total demand of all loads, respectively. Equation (III-3f) ensures that every vertex (except the source) is entered by exactly one edge.

Equations (III-4a) to (III-4c) limit the magnitude of voltage drops. Specifically, Equation (III-4a) follows Ohm's law and describes the voltage drop between two connected vertices  $i$  and  $j$ . According to Ohm's law, the voltage drop is proportional to the peak flow (i. e., the discounted sum of the peak demands). It is also proportional to the edge length  $l_{ij}$  and inversely proportional to the line's cross section (i. e., the edge capacity)  $a_{ij}^k$ . Equation (III-4b) demands that the voltage  $U_i$  of any vertex  $i$  cannot drop below a critical voltage level  $U_{\text{crit}}$ . Equation (III-4c) sets the voltage at the source to the nominal level  $U$ .

### 1.3 Complexity of the CAVLP

The following proposition prohibits straightforward solutions to the CAVLP due to its NP-hardness.

**Proposition 1 (NP-hardness)** *The CAVLP is NP-hard.*

Proposition 1 is proven by reduction. It is shown that the CAVLP is a generalized form of the problem in Brimberg et al. (2003), which is known to be NP-hard. More precisely, it is shown that the problem in Brimberg et al. (2003) is a special case of the CAVLP with  $U_{\text{crit}} = 0$  and a uniform coincidence factor  $\gamma \equiv 1$ .

By setting  $U_{\text{crit}} = 0$ , we can ignore the constraint for the voltage drops in Equations (III-4a) to (III-4c). We further set  $\gamma \equiv 1$ , which yields

$$\min \sum_{(i,j)^k \in E} x_{ij}^k [l_{ij}c_c + l_{ij}c_m a_{ij}^k] \quad (\text{III-5})$$

$$\text{s. t.} \quad \sum_{k \in \{1, \dots, |A|\}} x_{ij}^k a_{ij}^k \geq F_{ij}, \quad \forall i, j \in \{0, \dots, N-1\}, \quad (\text{III-6})$$

$$\sum_j F_{ji} - \sum_j F_{ij} = D_i \quad (\text{III-7a})$$

$$\sum_i \sum_k x_{ij}^k = 1, \quad \forall j \in \{1, \dots, N-1\}. \quad (\text{III-7b})$$

This problem is equivalent to the problem in Brimberg et al. (2003) with edge cost set to  $l_{ij}c_c + l_{ij}c_m a_{ij}^k$ . As this reduction is clearly of polynomial time, this proves that the CAVLP is NP-hard.  $\square$

Solving the CAVLP via complete enumeration is not feasible, even for smallest problem instances. This is expressed by the following remark.

**Remark 1** *The time for solving the CAVLP via complete enumeration is in  $O(N^{N-2})$ .*

*Proof.* For his proof, Cayley's theorem is used (Cayley, 1889). According to this theorem, there are  $N^{N-2}$  different trees for  $N$  vertices. To solve the CAVLP via complete

enumeration, one has to try all these combinations, and, in addition to that, enumerate through all possible combinations for the capacities. For a network with  $(N - 1)$  edges, there are  $|N|^{N-1}$  possible capacity permutations.  $\square$

Note that Equations (III-1) to (III-4c) give a mixed-integer nonlinear program. More precisely, nonlinearities are found in Equations (III-3a) to (III-3c), and Equation (III-4a). The quadratic nonlinearities in Equation (III-3b), (III-3c) and (III-4a) can be resolved by using the Big M method. The nonlinearities in Equation (III-3a) are more complex. The equation contains a product of the decision variable  $x_{ij}^k$  with the auxiliary decision variable  $\bar{D}_j$  and with the nonlinear function  $\gamma(|I_j|)$ , which depends on the auxiliary decision variable  $|I_j|$ .

As is shown later, the complexity of the CAVLP makes it intractable to use common commercial mixed-integer programming (MIP) solvers even for relatively small networks. For this reason, heuristics are developed later. A special case occurs in case of uniform coincidence factors  $\gamma \equiv 1$  throughout the network. Then, approximate solutions can be found using MIP solvers in a reasonable amount of time. This approach is later utilized to derive upper and lower bounds for the network cost. The upper bound is determined by choosing  $\gamma \equiv 1$ , while the lower bound is obtained by underestimating the coincidence factor, i. e.,  $\gamma \equiv \gamma(N - 1)$ , which corresponds to the maximum discount factor for a network of size  $N$  (i. e., a network with  $N - 1$  non-zero loads).

In this thesis, two ways of getting rid of these nonlinearities in Equation (III-3a) are presented. First, there exists an alternative problem formulation using arborescences. This formulation is given in Section 1.4. As we see below, in this formulation, the constraints grow exponentially in the number of vertices  $N$ . Therefore, this approach is not feasible. Second, in Section 1.5 (more precisely, in Section 1.5.1), a linearization of Equation (III-3a) is given. This linearization requires first a piecewise linearization of the coincidence factor  $\gamma(|I_j|)$ , after which the equation still contains a cubic non-

linearity. All linearizations of the equations mentioned above are provided below in Section 1.5.

## 1.4 Alternative Problem Formulation

### 1.4.1 Alternative Model

There is an alternative way of formulating the CAVLP. This alternative formulation is based on arborescence problems such as the CSSA<sub>r</sub> problem (see Chandy and Lo, 1973; Papadimitriou, 1978; Toth and Vigo, 1995) and avoids these nonlinearities. This formulation is given by

$$\min \sum_{(i,j)^k \in E} x_{ij}^k [l_{ij}c_c + l_{ij}c_m a_{ij}^k] \quad (\text{III-8})$$

$$\text{subject to} \quad \sum_{i \in V \setminus S} \sum_{j \in S} \sum_{k \in \{1, \dots, |A|\}} x_{ij}^k a_{ij}^k \geq \gamma(|S|) \sum_{j \in S} D_j, \quad \forall S \in \mathcal{P}(V \setminus \{0\}), \quad (\text{III-9})$$

$$x_{i0}^k = 0, \quad \forall i \in \{0, \dots, N-1\}, \quad \forall k \in \{1, \dots, |A|\}, \quad (\text{III-10a})$$

$$\sum_i \sum_k x_{ij}^k = 1, \quad \forall j \in V \setminus \{0\}, \quad (\text{III-10b})$$

$$\sum_{i \in V \setminus S} \sum_{j \in S} \sum_k x_{ij}^k \frac{a_{ij}^k}{l_{ij}} (U_i - U_j) = \gamma(|S|) \sum_{n \in S} D_n, \quad (\text{III-11a})$$

$$\forall S \in \mathcal{P}(V \setminus \{0\}), \quad \text{with} \quad \sum_{m \in S} \sum_{n \in V} \sum_k x_{mn}^k = |S| - 1,$$

$$U_i \geq U_{\text{crit}} = U - Q, \quad \forall i \in \{0, \dots, N-1\}, \quad (\text{III-11b})$$

$$U_0 = U. \quad (\text{III-11c})$$

The objective in Equation (III-8) is identical to Equation (III-1). Equation (III-9) follows the notation used in Toth and Vigo (1995) and ensures both connectivity of the graph and the line sizing constraint. Here,  $S$  is a subset of vertices and  $\mathcal{P}(V)$  is the powerset of  $V$ , that is, the set of all subsets of vertices. The line sizing constraint requires that the capacity  $a_{ij}^k$  of an edge  $(i, j)^k$  must be sufficiently large to support the peak flow,

i. e., the sum of all demands downstream of this edge, discounted by the coincidence factor  $\gamma(|S|)$ . The coincidence factor depends on the number of vertices in the vertex subset  $S$ .

Equation (III-10a) defines vertex 0 as the source, i. e., no edge is entering it. Equation (III-10b) ensures that every vertex (except the source) is entered by exactly one edge. Together with Equation (III-9), this ensures radiality.

Equations (III-11a) to (III-11c), again are identical to Equations (III-4a) to (III-4c) in the main problem formulation. Note that Equation (III-11a) is a quadratic inequality. However, this can be easily linearized using the Big M method.

#### 1.4.2 Complexity of the Alternative Problem Formulation

It can be shown that in the alternative problem formulation, the CAVLP is also NP-hard.

**Proposition 2 (NP-hardness)** *Using the alternative problem formulation in Equations (III-8) to (III-11c), the CAVLP is also NP-hard.*

The proof of this proposition can be found in Section 6.1.

The alternative problem formulation comes at the cost of the number of constraints scaling exponentially in the number of vertices  $N$ . This is because of the powersets in Equations (III-9) and (III-11a). Therefore, the alternative formulation is inferior from a computational standpoint.

### 1.5 Linearization of the CAVLP

Below, a linearization of the CAVLP is presented. The linearization of the flow constraint in Equation (III-3a) is most complex, as it requires first a piecewise linearization of the coincidence factor  $\gamma(|I_j|)$ , which then yields a cubic nonlinearity. This linearization is discussed in Section 1.5.1. Linearization of the quadratic nonlinearity in the voltage drop constraint in Equation (III-4a) is required to derive upper and

lower bounds for the problem. This linearization is discussed in Section 1.5.2. The remaining linearizations are discussed in Section 1.5.3.

### 1.5.1 Linearization of Flow Constraint

The first step in linearizing Equation (III-3a) is a piecewise linearization of the coincidence factor  $\gamma(|\Gamma_j|)$ . This is achieved by introducing a binary variable  $w_{jn}$ , together with the constraints

$$w_{jn} = \begin{cases} 1, & \text{if } |\Gamma_j| = n, \\ 0, & \text{otherwise,} \end{cases} \quad \forall j, n \in V, \quad (\text{III-12})$$

$$\sum_n w_{jn} = 1, \quad \forall j \in V. \quad (\text{III-13})$$

Next, we introduce a variable  $\gamma_j$ , with

$$\gamma_j = \sum_n \gamma(n)w_{jn}, \quad \forall j \in V. \quad (\text{III-14})$$

After this linearization of  $\gamma(|\Gamma_j|)$ , there is a cubic nonlinearity in Equation (III-3a), because of the product of three variables, namely  $x_{ij}^k$ ,  $w_{jn}$ , and  $\bar{D}_j$ .

We introduce a new variable  $z_{ijn}^k$ , representing the product  $z_{ijn}^k \stackrel{\text{def}}{=} x_{ij}^k w_{jn}$ . This is done by using the inequalities

$$z_{ijn}^k \leq x_{ij}^k, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-15a})$$

$$z_{ijn}^k \leq w_{jn}, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-15b})$$

$$z_{ijn}^k \geq x_{ij}^k + w_{jn} - 1, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}. \quad (\text{III-15c})$$

Analogously, we introduce a second variable  $z'_{ijn}$  for the product  $z'_{ijn} \stackrel{\text{def}}{=} x_{ij}^k w_{in}$  (the index  $i$  is used instead of  $j$  in the multiplication). This yields

$$z'_{ijn} \leq x_{ij}^k, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-16a})$$

$$z'_{ijn} \leq w_{in}, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-16b})$$

$$z'_{ijn} \geq x_{ij}^k + w_{in} - 1, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}. \quad (\text{III-16c})$$

In the last step, we are left with the quadratic nonlinearity resulting from the products  $z_{ijn}^k \bar{D}_j$  and  $z'_{ijn} \bar{D}_j$ . This can be resolved via the Big M method. We introduce two new variables  $\Delta_{ijn}^k \stackrel{\text{def}}{=} z_{ijn}^k \bar{D}_j$  and  $\Delta'_{ijn} \stackrel{\text{def}}{=} z'_{ijn} \bar{D}_j$ , together with

$$\Delta_{ijn}^k \leq z_{ijn}^k M, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-17a})$$

$$\Delta_{ijn}^k \geq 0, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-17b})$$

$$\Delta_{ijn}^k \leq \bar{D}_j, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-17c})$$

$$\Delta_{ijn}^k \geq \bar{D}_j - (1 - z_{ijn}^k)M, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-17d})$$

as well as

$$\Delta'_{ijn} \leq z'_{ijn} M, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-18a})$$

$$\Delta'_{ijn} \geq 0, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-18b})$$

$$\Delta'_{ijn} \leq \bar{D}_j, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-18c})$$

$$\Delta'_{ijn} \geq \bar{D}_j - (1 - z'_{ijn})M, \quad \forall i, j, n \in V, \forall k \in \{1, \dots, |A|\}. \quad (\text{III-18d})$$

As a result, we arrive at the linearized version of Equation (III-3a), which reads

$$\sum_j F_{ji} - \sum_j F_{ij} = \gamma_i D_i - \sum_j \sum_k \left( \sum_n \gamma(n) \Delta_{ijn}^k - \sum_n \gamma(n) \Delta'_{ijn} \right), \quad \forall i \in V \setminus \{0\}.$$

(III-19)

### 1.5.2 Linearization of Voltage Drop Constraint

In the following, the quadratic constraint representing Ohm's law in Equation (III-4a) are linearized using the Big M notation. In doing so, the approach of Avella et al. (2005) is followed. Thus, Equation (III-4a) is replaced with two linear inequalities. This yields

$$\frac{a_{ij}^k}{l_{ij}}(U_i - U_j) \leq F_{ij} + (1 - x_{ij}^k)M, \quad \forall i, j \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-20})$$

$$\frac{a_{ij}^k}{l_{ij}}(U_i - U_j) \geq F_{ij} - (1 - x_{ij}^k)M, \quad \forall i, j \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-21})$$

where  $M$  is a suitable large number. If an edge is selected (i. e., if  $x_{ij}^k = 1$ ), the combination of Equations (III-20) to (III-21) is equivalent to Equation (III-4a). If  $x_{ij}^k = 0$ , the equations become redundant.

### 1.5.3 Linearization of Remaining Nonlinearities

Equations (III-3b) and (III-3c) is linearized by introducing two new variables  $g_{ij}^k$  and  $d_{ij}^k$ , representing the products  $g_{ij}^k \stackrel{\text{def}}{=} x_{ij}^k |\Gamma_j|$  and  $d_{ij}^k \stackrel{\text{def}}{=} x_{ij}^k \overline{D}_j$ . These products can be written down in a linear way by using the Big M method. This yields

$$g_{ij}^k \leq x_{ij}^k M, \quad \forall i, j \in V \setminus \{0\}, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-22a})$$

$$g_{ij}^k \geq 0, \quad \forall i, j \in V \setminus \{0\}, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-22b})$$

$$g_{ij}^k \leq |\Gamma_j|, \quad \forall i, j \in V \setminus \{0\}, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-22c})$$

$$g_{ij}^k \geq |\Gamma_j| - (1 - x_{ij}^k)M, \quad \forall i, j \in V \setminus \{0\}, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-22d})$$

as well as

$$d_{ij}^k \leq x_{ij}^k M, \quad \forall i, j \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-23a})$$

$$d_{ij}^k \geq 0, \quad \forall i, j \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-23b})$$

$$d_{ij}^k \leq \bar{D}_j, \quad \forall i, j \in V, \forall k \in \{1, \dots, |A|\}, \quad (\text{III-23c})$$

$$d_{ij}^k \geq \bar{D}_j - (1 - x_{ij}^k) M, \quad \forall i, j \in V, \forall k \in \{1, \dots, |A|\}. \quad (\text{III-23d})$$

$M$  is a suitable large number. With these equations, the linearized forms of Equations (III-3b) and (III-3c) read

$$|F_i| = 1 + \sum_j \sum_k g_{ij}^k, \quad \forall i \in V, \quad (\text{III-24a})$$

$$\bar{D}_i = D_i + \sum_j \sum_k d_{ij}^k, \quad \forall i \in V. \quad (\text{III-24b})$$

## 1.6 Solution Properties

In the following section, four properties of the optimal solution of the CAVLP are presented. The first two properties describe the network layout of the optimal solution, while properties three and four characterize the capacities. These properties are used later to develop the solution methods.

### 1.6.1 Optimal Network Layout for Low Demand Situations

The minimum spanning tree (MST) is the cycle-free network connecting all vertices with the shortest total edge length (Prim, 1957). Let  $\Gamma^{\text{MST}}$  denote a solution of the CAVLP with an MST layout.

**Remark 2** For  $D_i \rightarrow 0$  or  $a_{ij}^k \rightarrow \infty$ , the MST layout is the optimal solution to the CAVLP.

*Proof.* When  $D_i \rightarrow 0$ , we find that  $F_{ij} \rightarrow 0$ . Consequently, the constraints for both line sizing (Equation (III-2)) and voltage drops (Equation (III-4b)) are fulfilled for any

choice of  $x_{ij}^k$ . The same holds true for  $a_{ij}^k \rightarrow \infty$ . All capacities can be set to the lowest possible value  $a_{\min} = \min_k a_{ij}^k$ . Then, the objective function in Equation (III-1) simplifies to  $\min \sum_{i,j} x_{ij} l_{ij} [c_c + c_m + a_{\min}]$ . The term  $[c_c + c_m + a_{\min}]$  is constant and, therefore, the objective function is identical to the objective function of the MST problem.  $\square$

### 1.6.2 Optimal Network Layout for High Demand Situations

In situations with relatively high demand, many network layouts will not yield feasible solutions, even when all capacities are set to the maximum possible value. Let  $\Gamma^{\text{Star}}$  (“starred network”) denote a solution of the CAVLP in which every vertex is connected directly to the source 0, i. e.,  $x_{ij} = 1$  for  $i = 0$  and, otherwise,  $x_{ij} = 0$ .

**Remark 3** *If feasible solutions to the CAVLP exist, the starred network is one of these solutions.*

*Proof.* This remark is proven by contradiction. Consider a starred network  $\Gamma^1$  and assume that this solution violates one of the constraints in Equation (III-2) or Equation (III-4b) for the sub-tree consisting of only the edge connecting the vertex  $v$  to the source 0. We further assume that there exists an alternative solution  $\Gamma^2$  not violating the constraints, and that in this solution vertex  $v$  is connected to a vertex  $w$  other than 0. Without loss of generality, we assume that all capacities in both the starred layout and the alternative layout are set to the maximum value  $a_{\max} = \max_k a_{ij}^k$ .

We distinguish two cases. First, we consider the line sizing constraint in Equation (III-2). It is obvious that the line sizing constraint cannot be the reason why  $\Gamma^2$  is feasible and  $\Gamma^1$  is not, since  $F_{uw}^2 > F_{0v}^1 = D_v$  and  $D_i \in \mathbb{R}^+$ , i. e., the flows on the edges in the starred network are minimal. It should be noted, that we require the coincidence factor to have a form such that adding demands to an edge always increases the flows, i. e.,  $\frac{\gamma(|I_j|+1)}{\gamma(|I_j|)} > \frac{|I_j|}{|I_j|+1}$ . This is the case for all forms that can be found in the literature (Dickert and Schegner, 2010). Second, we focus on the constraint for the voltage drops in Equation (III-4b). By use of the triangle inequality, we show that

the total length of all edges from the source to the vertex in  $\Gamma^2$  is  $L^2 \geq l_{0v} = L^1$ . Therefore, Equation (III-4b) must have been fulfilled in the starred network  $\Gamma^1$  also. This contradicts the initial assumption and concludes the proof.  $\square$

### 1.6.3 Line Capacity Ratios

When allowing the choice of continuous capacities (i. e.,  $A = \mathbb{R}^+$ ), further properties with respect to the voltage drop can be derived. As we see later, the following proposition proves to be very powerful for developing efficient heuristics. In particular, it is beneficial if the constraint for the voltage drops is binding, which is typically the case in real-world applications. For better readability, the superscript  $k$  for the capacities is dropped. The capacity of the edge  $(i, j)^k$  are simply referred to by  $a_{ij}$ .

**Proposition 3 (Line capacity ratios)** *For any given network layout and for continuous capacities (i. e.,  $A = \mathbb{R}^+$ ), the capacities that minimize the cost fulfill*

$$\frac{a_{ij}^2}{a_{mn}^2} = \frac{F_{ij}}{F_{mn}} \quad \text{or} \quad \frac{a_{ij}}{a_{mn}} = \sqrt{\frac{F_{ij}}{F_{mn}}} \quad (\text{III-25})$$

for any two edges  $(i, j)$  and  $(m, n)$  in the same sub-tree and if Equation (III-4b) is binding.

*Proof.* We assume that the set of possible capacities is continuous and all capacities can take up any real value. As a first step, consider a very simple network consisting of 3 vertices (with  $i = 0, 1, 2$ ) and 2 edges. Later, we expand this to networks of arbitrary length and branching. In the first step, one edge connects vertex 1 to the source. Vertex 2 is connected to Vertex 1 by a second edge. Using the objective function in Equation (III-1), we can set up the cost function for this network as

$$C = l_{01} c_c + l_{01} c_m a_{01} + l_{12} c_c + l_{12} c_m a_{12} . \quad (\text{III-26})$$

For a given solution to the CAVLP, we derive the accumulated voltage drop from Equation (III-40), which yields

$$Q = l_{01} \frac{F_{01}}{a_{01}} + l_{12} \frac{F_{12}}{a_{12}} . \quad (\text{III-27})$$

To minimize the cost function in Equation (III-26), we derive the Lagrangian for this problem, which yields

$$\mathcal{L} = c_c(l_{01} + l_{12}) + c_m(l_{01} a_{01} + l_{12} a_{12}) + \lambda \left( l_{01} \frac{F_{01}}{a_{01}} + l_{12} \frac{F_{12}}{a_{12}} - Q \right) , \quad (\text{III-28})$$

where  $\lambda$  is the Lagrange multiplier to include the voltage drop from Equation (III-27). We then take the partial derivatives with respect to  $a_{01}$ ,  $a_{12}$  and  $\lambda$ . By setting them to zero, we arrive at the following system of equations:

$$\frac{\partial \mathcal{L}}{\partial a_{01}} = l_{01} c_m - \lambda l_{01} \frac{F_{01}}{(a_{01})^2} = 0 , \quad (\text{III-29})$$

$$\frac{\partial \mathcal{L}}{\partial a_{12}} = l_{12} c_m - \lambda l_{12} \frac{F_{12}}{(a_{12})^2} = 0 , \quad \text{and} \quad (\text{III-30})$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \frac{l_{01} F_{01}}{a_{01}} + \frac{l_{12} F_{12}}{a_{12}} - Q = 0 . \quad (\text{III-31})$$

In Equations (III-29) and (III-30), the lengths cancel out and the two formulas can be rewritten to

$$c_m - \lambda \frac{F_{01}}{a_{01}^2} = 0 , \quad \text{and} \quad (\text{III-32})$$

$$c_m - \lambda \frac{F_{12}}{a_{12}^2} = 0 . \quad (\text{III-33})$$

From Equations (III-32) and (III-33), a generalized formula for sub-trees of arbitrary length can be derived. The generalized formula is

$$c_m - \lambda \frac{F_{ij}}{a_{ij}^2} = 0 , \quad (\text{III-34})$$

which can be rewritten to

$$a_{ij}^2 = \frac{\lambda}{c_m} F_{ij}. \quad (\text{III-35})$$

The square of the capacity of an edge  $a_{ij}^2$  is proportional to the flow  $F_{ij}$ . For any two edges  $(i, j)^k$  and  $(m, n)^k$  of the same sub-tree, we find

$$\frac{a_{ij}^2}{a_{mn}^2} = \frac{F_{ij}}{F_{mn}}. \quad (\text{III-36})$$

This concludes the proof.  $\square$

#### 1.6.4 Decreasing line capacities

The following corollary states that, starting from the source to the leaves within the network, line capacities are monotonically decreasing. This follows directly from Proposition 3. Formally, the flow  $F_{ij}$  through an edge  $(i, j)^k$  is the sum of all demands downstream to this edge. All demands are positive real numbers, i. e.,  $D_i \in \mathbb{R}^+$ , for all  $i$ . Therefore, the flows are decreasing when moving downstream.

**Corollary 1** *Under the above assumptions, the cost-minimizing line capacities for any given network layout are decreasing when moving downstream.*

## 2 Optimization Methods

The CAVLP problem is solved by dividing it into two sub-problems: (A) generating the network layout and (B) capacity optimization. Generating the network layout (A) is addressed by two sets of heuristics. These first create an initial solution (Section 2.1) and, given an initial solution as input, then make local improvements to the network layout (Section 2.2). For improvements to the network layout, we also present two metaheuristics, namely variable neighborhood search and Tabu Search. Capacity optimization (B) is addressed via heuristics and exact solvers in order to determine line

type of each edge for a given layout (Section 2.3). The choice of capacity optimization method is independent from the layout generation and improvement methods.

## 2.1 Generating the Initial Network Layout

The objective of generating an initial network is to create a network layout (without considering different line types). To this end, multigraph  $G$  is reduced to a graph  $\tilde{G} = (V, \tilde{E})$  whereby multiple edges between the same vertices are replaced by a single edge, that is, by setting  $A = \{a\}$ . This affects the optimization problem in Equations (III-1) to (III-4c) analogously. Here the index  $k$  can be dropped and, thus, the decision variable becomes  $x_{ij} \in \{0, 1\}$ . The decision variable  $x_{ij}$  indicates whether an edge  $(i, j) \in \tilde{E}$  from vertex  $i$  to vertex  $j$  should be built. The resulting network layout is represented by  $\{x_{ij}\}$ . Below, three methods for generating an initial network layout are presented. These layouts vary in their branching. The minimum spanning tree algorithm creates layouts with low branching. The layouts generated by the greedy network construction are slightly more branched as they are more centered around the source. The networks resulting from the starred network algorithm exhibit a high degree of branching because all edges are directly connected to the source.

### 2.1.1 Minimum Spanning Tree Algorithm

The MST connects all vertices, so that total length of all edges is minimized. In Remark 2, it has been shown that the MST is the optimal solution to the CAVLP for situations in which voltage drops can be neglected. This makes the MST favorable for instances with low demands  $D_i$ . In the implementation of the MST used in this thesis, Prim's algorithm is used (Prim, 1957). A detailed description of the algorithm is provided in Algorithm 1.

The algorithm subsequently adds vertices to an existing graph. Thereby, in each iteration, the edge adding the least cost (i. e., the shortest length) is chosen to connect new vertices. The vertices yet to be added to the graph are stored in a list  $Q$  (line 2).

The algorithm uses so-called key values  $k_j$  that represent the smallest cost for adding a vertex  $j$  to the existing graph. For each vertex  $j$ , the list  $e_j$  stores the best option for  $j$  to be connected to. The two lists are initialized in line 3. In line 4, the algorithm sets the key value for vertex 0, so that it is added first to the graph. Note that this choice is arbitrary. In lines 5 to 5, the algorithm adds vertices, until every vertex is connected. This works as follows. The algorithm picks the vertex that adds the least cost in line 6. This vertex is removed from the list of disconnected vertices (line 7). The edge connecting this vertex is added in line 8. This step is skipped in the first iteration. Next, the algorithm updates the key values in lines 10 to 12 by checking, if the newly added vertex opened up less expensive ways of adding further vertices.

---

**Algorithm 1** Prim's Algorithm
 

---

```

1: Initialize  $x_{ij} \leftarrow 0$ , for all  $i, j$ 
2:  $Q \leftarrow V$ 
3: Initialize  $k_j \leftarrow \infty$ ,  $e_j \leftarrow \{\}$ , for all  $j$ 
4:  $k_0 \leftarrow 0$ 
5: while  $Q \neq \{\}$  do
6:   Pick vertex  $j$  with  $k_j = \min_i \{k_i\}$ 
7:    $Q \leftarrow Q \setminus j$ 
8:   if  $v \neq 0$  then
9:      $i \leftarrow e_j$ ,  $x_{ij} \leftarrow 1$ 
10:  for each vertex  $m$  do
11:     $k_m \leftarrow \min\{l_{jm}, k_m\}$ 
12:    Set  $e_m$  gets  $j$ , if  $k_m$  was updated
13: return  $\{x_{ij}\}$ 

```

---

Prim's algorithm has a runtime of  $O((N - 1) \log N)$ .

### 2.1.2 Starred Network Algorithm

In a starred network, every vertex is connected directly to the source 0. Thus, the starred network algorithm sets  $x_{0j} \leftarrow 1$  for all  $j$  and  $x_{ij} \leftarrow 0$  otherwise. The starred network layout minimizes the flow through the edges and is thus favorable for instances with high demands  $D_i$  (see also Remark 3).

### 2.1.3 Greedy Network Construction

A greedy heuristic for network construction is developed in the following. It adds vertices to the network in the increasing order of their distance to the source. Thereby,

this heuristic creates layouts that have a longer total edge length but are more centered around the source than the MST (where the order in which vertices are added depends on their distance to each other). The pseudocode is provided in Algorithm 2. The greedy network construction uses two data structures with vertices, which are initialized in line 2: a set  $C$  contains vertices that are already connected to the network (at the beginning, this list contains only the source), while  $D$  contains vertices that have not yet been added to the network. In line 3, the heuristic sorts all vertices  $j \in D$  by their distance to the source  $l_{0j}$ . In lines 4 to 6, it iteratively adds these vertices to the existing network by the shortest connection possible. Owing to line 5, the heuristic has a runtime of  $O(N^2)$ .

---

**Algorithm 2** Greedy network construction

---

```

1: Initialize  $x_{ij} \leftarrow 0$  for all  $i, j$ 
2: Set  $C \leftarrow \{0\}$  and  $D \leftarrow [1, \dots, N - 1]$ 
3: Sort vertices  $j$  in  $D$  by  $l_{0j}$  (ascending)
4: for each vertex  $v \in D$  do
5:    $x_{z^*,v} \leftarrow 1$  where  $z^* = \arg \min_{z \in C} l_{zv}$ 
6:    $C \leftarrow C \cup v$ 
7: return  $\{x_{ij}\}$ 

```

---

## 2.2 Improving an Existing Network Layout

Departing from these initial layouts, several heuristics to improve the layout are developed, namely the increased network branching heuristics, the decreased network branching heuristic, and the randomized network reconfiguration. These heuristics take a given layout  $\{x_{ij}\}$  as input and then subsequently modify it. The objective is to improve the overall network cost, i. e., the objective function in Equation (III-1). To calculate the overall network cost, the algorithms rely upon additional input in form of the line capacities  $\{a_{ij}\}$ . These are determined in the second sub-problem, capacity optimization, which is presented later in Section 2.3. In the following heuristics, a routine for capacity optimization (i. e., CAPACITYOPTIMIZATIONMETHOD) is called, where the capacities  $\{a_{ij}\}$  should enter the algorithm. The CAPACITYOPTIMIZATIONMETHOD returns a suggested set of capacities  $\{a_{ij}\}$  for a given layout  $\{x_{ij}\}$ .

### 2.2.1 Randomized Network Reconfiguration

The randomized network reconfiguration works by randomly adding an edge  $(i, j)$  to a given network layout. This forms a cycle  $C$ . The algorithm then attempts to remove each edge inside the cycle as means to restore the radial layout. In doing so, it checks if a radial network is found that improves cost compared to the original one. The pseudocode is provided in Algorithm 3. In line 1, the capacities are determined. These capacities are used in line 2 to calculate the cost  $c^*$  of the initial layout. The initial layout  $\{x_{ij}\}$  is stored as  $X^*$  in line 3. In line 4, the heuristic alters the network layout. This process is repeated for a pre-defined number of  $s_{\max}$  iterations. In lines 5 to 6, the heuristic adds a new edge  $(i, j)$  selected at random by setting  $x_{ij} \leftarrow 1$ . The resulting cycle  $C$  is determined in line 7. In line 8, the algorithm iterates over each  $(p, q) \in C$  that is not the newly added edge. The edge  $(p, q)$  is deleted in line 9 to obtain a radial layout. This may require the direction of some edges to be reversed in line 10. The capacities  $\{a_{ij}\}$  and cost  $c_s$  for the new network layout are determined in lines 11 to 12. If the cost has improved than the current best solution, the algorithm saves the current network as the new best solution in line 13. In lines 14 to 15, the changes are reversed before proceeding with the next iterations. After  $s_{\max}$  iterations, the heuristic returns the network with the lowest cost (line 16).

---

#### Algorithm 3 Randomized network reconfiguration

---

**Input:** Network layout  $\{x_{ij}\}$

```

1:  $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$ 
2:  $c^* \leftarrow \text{COST}(\{x_{ij}\}, \{a_{ij}\})$  ▷ Determine initial cost and save it as current best cost
3:  $X^* \leftarrow \{x_{ij}\}$ 
4: for  $s \in \{1, \dots, s_{\max}\}$  do
5:   Randomly select disabled edge  $(i, j)$ , with  $x_{ij} = x_{ji} = 0$ 
6:    $x_{ij} \leftarrow 1$  ▷ Add edge  $(i, j)$ 
7:   Compute list  $C$  containing all edges comprising the cycle
8:   for each edge  $(p, q) \neq (i, j) \in C$  do
9:      $x_{pq} \leftarrow 0$  ▷ Delete edge inside the cycle to create radial layout
10:    Check direction of all edges  $(m, n) \in C$  and reverse if necessary
11:     $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$ 
12:     $c \leftarrow \text{COST}(\{x_{ij}\}, \{a_{ij}\})$  ▷ Determine new cost
13:    if  $c < c^*$  then  $c^* \leftarrow c$  and  $X^* \leftarrow \{x_{ij}\}$  ▷ Update network if cost is cheaper
14:     $x_{pq} \leftarrow 1$  ▷ Close cycle to reset old configuration
15:     $\{x_{ij}\} \leftarrow X^*$ 
16: return  $X^*$ 

```

---

## 2.2.2 Increased Network Branching Heuristic

The principle of the increased network branching (INB) heuristic is to reconnect a subgraph  $\tilde{T}_j$  to a vertex of lower depth such as the source or a vertex in the vicinity of the source. This reduces the depth of the network and increases the branching. By disconnecting  $\tilde{T}_j$  from a subgraph, material cost in this subgraph can be saved because the flows and voltage drops are getting smaller. These cost savings need to be compared to the additional cost for reconnecting  $\tilde{T}_j$  to a different vertex of the network. The pseudocode is provided in Algorithm 4. In lines 1 and 2, the capacities are determined and the cost of the initial layout is calculated. In line 3, the currently cheapest network layout is stored in  $X^*$ . The heuristic now loops over all vertices  $j$  of a certain depth  $d(j)$  up to a pre-defined maximum depth  $d_{\max}$ . In line 5, the heuristic disconnects the vertex  $j$ , resulting in two subgraphs:  $\tilde{T}_0$  and  $\tilde{T}_j$ . In lines 7 to 15, the heuristic loops over all vertices  $i \in \tilde{T}_0$  with depth  $d(i) < d$  and all vertices in  $\tilde{T}_j$  in order to evaluate potential cost reductions. In line 16, the heuristic returns the cheapest layout.

---

### Algorithm 4 Increased network branching heuristic

---

**Input:** Network layout  $\{x_{ij}\}$

```

1:  $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$ 
2:  $c^* \leftarrow \text{COST}(\{x_{ij}\}, \{a_{ij}\})$  ▷ Determine initial cost
3:  $X^* \leftarrow \{x_{ij}\}$ 
4: for each vertex  $j$  with  $d(j) \in \{2, \dots, d_{\max}\}$  do
5:    $x_{ij} \leftarrow 0$  ▷ Delete edge
6:   Compute subgraph  $\tilde{T}_j$  without source
7:   for each vertex  $p \notin \tilde{T}_j$  with  $d(p) < d$  do
8:     for vertex  $q \in \tilde{T}_j$  do
9:        $x_{pq} \leftarrow 1$  ▷ Reconnect  $\tilde{T}_j$ 
10:      for each edge  $(m, n) \in \tilde{T}_j$  in path from  $j$  to  $q$  do
11:         $x_{mn} \leftarrow 0, x_{nm} \leftarrow 1$  ▷ Check direction of reconnected edges
12:         $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$ 
13:         $c \leftarrow \text{COST}(\{x_{ij}\}, \{a_{ij}\})$  ▷ Determine new cost
14:        if  $c \leq c^*$  then  $X^* \leftarrow \{x_{ij}\}$  and  $c^* \leftarrow c$  ▷ Restore old layout if too expensive
15:        else  $\{x_{ij}\} \leftarrow X^*$ 
16: return  $X^*$ 

```

---

The runtime of this heuristic depends on the network layout, in particular on the branching in the vicinity of the source. It further scales linearly with the runtime of the capacity optimization method in line 12.

### 2.2.3 Decreased Network Branching Heuristic

The decreased network branching (DNB) heuristic works in the opposite direction to the INB heuristic. It reconnects subgraph  $\tilde{T}_j$  with  $j$  of low depth to existing vertices  $q$  with a higher depth. The DNB heuristic works as follows. Up to a predefined threshold of maximum depth  $d_{\max}$ , the DNB heuristic deletes edges  $(i, j)$  with  $d(j) \leq d_{\max}$  (i. e., vertices relatively close to the source are disconnected). Again, this results in the subgraphs  $\tilde{T}_0$  and  $\tilde{T}_j$ . The subgraph  $\tilde{T}_j$  is then reconnected to a vertex  $q \in \tilde{T}_0$  with  $d(j) \leq d(q) \leq d_{\max}$ . If the resulting network is cheaper than the previous one, the new layout is kept; otherwise, the old layout is restored.

### 2.2.4 Variable Neighborhood Search Metaheuristic

The idea behind VNS is to change neighborhoods in order to find a better solution, as opposed to local search methods that do not use several neighborhoods within one method (cf. Hansen and Mladenović, 2014; Hansen et al., 2019; Mladenović and Hansen, 1997). For the change in neighborhoods, a metric for the distance between solutions needs to be introduced. In our case, the distance between two layouts is defined by their difference in edges. For example, a certain layout  $A$  has distance 1 to another layout  $B$  if it can be reached by adding just one edge to  $B$  and deleting another. We then say that that  $A$  is part of the “1-neighborhood” of  $B$ .

VNS has two main components, which are (a) shaking (i. e., the change of neighborhoods) and (b) local search. The pseudocode is provided in Algorithm 5. In lines 1 to 3, the heuristic determines capacities and initial cost and saves the initial layout. The VNS procedure starts in line 4 and runs for  $s_{\text{VNS}}$  iterations. In line 5, the distance  $d$  is set to 1. In the first iteration, the algorithm thus starts exploring the 1-neighborhood of the initial solution. The algorithm explores neighborhoods in a distance of up to  $d_{\max}$  (line 6). Lines 7 and 8 state the shaking procedure. Here a method called intensified shaking is used, where, instead of conventional shaking by drawing an arbitrary neighboring layout, a more strategic procedure is applied (see, e. g., Brimberg et al., 2003; Hansen and Mladenović, 2014). For this purpose, the randomized network re-

configuration method is used with just one iteration (i. e.,  $s_{\max} = 1$ ) to choose one edge that is to be added at random and then find the best edge to be removed. In line 9, the local search is performed. The cost for the best layout found is determined in lines 10 to 11. If the cost is smaller than the current best solution, the algorithm saves the new best solution in lines 12 to 14 and proceeds with the next iteration. Otherwise, the algorithm increases  $d$  to explore the next neighborhood in line 15. At the end, the heuristic returns the cheapest network in line 16.

---

**Algorithm 5** Variable Neighborhood Search

---

**Input:** Network layout  $\{x_{ij}\}$

```

1:  $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$ 
2:  $c^* \leftarrow \text{COST}(\{x_{ij}\}, \{a_{ij}\})$  ▷ Determine initial cost
3:  $X, X^* \leftarrow \{x_{ij}\}$ 
4: for  $s \in \{1, \dots, s_{\text{VNS}}\}$  do
5:    $d \leftarrow 1$  ▷ Set distance to 1
6:   while  $d \leq d_{\max}$  do
7:     for  $d$  times do ▷ Intensified shaking
8:        $X \leftarrow \text{RANDOMIZEDNETWORKRECONFIGURATION}(\{x_{ij}\}, s_{\max} = 1)$  ▷ Draw neighboring layout
9:       Perform Local Search
10:       $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$ 
11:       $c \leftarrow \text{COST}(\{x_{ij}\}, \{a_{ij}\})$  ▷ Determine new cost
12:      if  $c \leq c^*$  then
13:         $X, X^* \leftarrow \{x_{ij}\}$  and  $c^* \leftarrow c$  ▷ Update best solution if cost has improved
14:        break ▷ Return to line 5
15:      else  $d \leftarrow d + 1$  and  $\{x_{ij}\} \leftarrow X$  ▷ Increase distance to explore next neighborhood
16: return  $X^*$ 

```

---

The CAVLP allows edges between any two vertices. This makes the objective function very sensitive to the shaking procedure, i. e., a wrong choice of an edge to be added might strongly deteriorate the objective value regardless of which edge is subsequently removed. Therefore, in the experiments presented here, the VNS algorithm is modified to achieve better results. This is done by further intensifying the neighborhood change via shaking in lines 7 and 8. This yields better computational results if one further adaptation is made to the shaking procedure described in lines 7 and 8. Instead of shaking only once (as described in Algorithm 5), line 8 is repeated five times and then select the best solution for the neighborhood change.

### 2.2.5 Tabu Search Metaheuristic

Tabu Search is a metaheuristic that uses short-term memory in the form of a tabu list to find solutions more efficiently (Gendreau and Potvin, 2014; Glover, 1989). The implementation of Tabu Search shown in this thesis uses a tabu list  $T$  of fixed length  $l_T$ . The list contains both recently deleted edges to avoid cycling back to previous solutions and unfavorable edges, i. e., edges that were recently explored without leading to improvements (note that experiments with two separate tabu lists have also been conducted but this did not yield better results).

---

#### Algorithm 6 Tabu Search

---

**Input:** Network layout  $\{x_{ij}\}$

- 1:  $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$
- 2:  $c^* \leftarrow \text{COST}(\{x_{ij}\}, \{a_{ij}\})$  ▷ Determine initial cost
- 3:  $X^* \leftarrow \{x_{ij}\}$
- 4:  $T \leftarrow \emptyset$  ▷ Initialize tabu list
- 5: **for**  $s \in \{1, \dots, s_{\text{Tabu}}\}$  **do**
- 6:   Select random disabled edge  $(i, j) \notin T$ , such that  $x_{ij} = x_{ji} = 0$
- 7:    $x_{ij} \leftarrow 1$  ▷ Add edge  $(i, j)$
- 8:    $t \leftarrow (i, j)$  ▷  $(i, j)$  is the candidate for the tabu list
- 9:   Compute list  $C$  containing all edges comprising the cycle
- 10:   **for** each edge  $(p, q) \neq (i, j) \in C$  **do**
- 11:      $x_{pq} \leftarrow 0$  ▷ Delete edge inside the cycle to create radial layout
- 12:     Check direction of all edges  $(m, n) \in C$  and reverse if necessary
- 13:      $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$
- 14:      $c \leftarrow \text{COST}(\{x_{ij}\}, \{a_{ij}\})$  ▷ Determine new cost
- 15:     **if**  $c < c^*$  **then**
- 16:        $c^* \leftarrow c$  and  $X^* \leftarrow \{x_{ij}\}$  ▷ Update network if cost is cheaper
- 17:        $t \leftarrow (p, q)$  ▷  $(p, q)$  is the new candidate for the tabu list
- 18:        $x_{pq} \leftarrow 1$  ▷ Close cycle to reset old configuration
- 19:      $\{x_{ij}\} \leftarrow X^*$
- 20:      $T \leftarrow T \cup t$  ▷ Update tabu list
- 21:     **if**  $|T| > l_T$  **then** remove first element in  $T$
- 22: **return**  $X^*$

---

The pseudocode of the Tabu Search algorithm is provided in Algorithm 6. In lines 1 to 3, the heuristic determines capacities and initial cost. In line 4, the tabu list  $T$  is initialized to an empty list. The main part of the algorithm starts in line 5 and is executed for a pre-defined number of  $s_{\text{Tabu}}$  iterations. The algorithm adds a random edge  $(i, j)$  in lines 6 to 7. This edge must not be in the tabu list  $T$ . It is then saved as a potential candidate for the tabu list in line 8. This means that if no better solution is found during the following process,  $(i, j)$  should not be added for the next iterations

because it is unfavorable. Lines 9 to 19 resemble the optimization procedure of the randomized network reconfiguration from Section 2.2.1 with one exception: if the algorithm finds a cheaper layout by deleting an edge  $(p, q)$ , this edge is then saved as the new candidate for the tabu list in line 17 to avoid cycling back to the previous solution. After each iteration, the tabu list is then updated in lines 20 to 21. Finally, the heuristic returns the cheapest network in line 22.

## 2.3 Optimizing the Capacities

The CAVLP is divided into two sub-problems: (A) generating the network layout and (B) capacity optimization. Capacity optimization (B) takes a network layout  $\{x_{ij}\}$  as input and determines the corresponding line types  $k$ . The main difference to the full CAVLP (i. e., the combination of the two sub-problems) is the fact that capacity optimization only considers a subset of edges  $E' \subset E$ , namely these edges where  $x_{ij} = 1$  has been determined in the first sub-problem. As a consequence, the flows are now given and  $F_{ij}$  is no longer an auxiliary decision variable. This reduces the complexity of the problem. Capacity optimization is formulated as a binary integer problem:

$$\min \sum_{(i,j)^k \in E'} x_{ij}^k l_{ij} a_{ij}^k \quad (\text{III-37})$$

$$\text{subject to } \sum_{k \in \{1, \dots, |A|\}} x_{ij}^k a_{ij}^k \geq F_{ij}, \quad \forall i, j \in \{0, \dots, N-1\}, \quad (\text{III-38})$$

$$\sum_k x_{ij}^k = 1, \quad \forall (i, j)^k \in E', \quad (\text{III-39})$$

$$\sum_{(i,j)^k \in P} x_{ij}^k l_{ij} \frac{F_{ij}}{a_{ij}^k} \leq U - U_{\text{crit}} = Q, \quad \forall p \in P. \quad (\text{III-40})$$

Since the layout is given, the objective function in Equation (III-37) only takes into account the material cost. For the same reason, the line sizing constraint can be simplified to Equation (III-38). Equation (III-39) ensures that exactly one line type is chosen for each connection. The constraint for the voltage drop is reformulated in

Equation (III-40). The equation demands that the sum of all voltage drops in any path  $p \in P$  from the source to a leaf must stay below the threshold  $Q = U - U_{\text{crit}}$ .  $P$  is the set of all paths from the source to a leaf in  $\Gamma$ . Because all paths  $P$  are given (due to the given layout), this is a very efficient formulation. The aforementioned problem for capacity optimization is approached below by a MIP solver for obtaining exact solutions and a heuristic.

### 2.3.1 Exact Solution with MIP Solver

The problem stated in Equations (III-37) to (III-40) is implemented as a binary integer problem and solved using a commercially available MIP solver, namely Gurobi Optimizer 7.5.2 with all parameters set to the default values (Gurobi Optimization, 2017).

### 2.3.2 Pairwise Edge Capacity Adjustment

The pairwise edge capacity adjustment (PECA) heuristic is based on two previously derived properties: first, Corollary 1 states that the farther away from the source, the smaller the capacities become. Second, the heuristic utilizes Proposition 3, which defines the optimal ratio between two line capacities in the same path  $p$  based on the flows, i. e.,  $\frac{a_{ij}}{a_{mn}} = \sqrt{\frac{F_{ij}}{F_{mn}}}$ . The PECA heuristic adjusts the capacities of two edges simultaneously in order to bring the ratio of these capacities as closely as possible to  $\sqrt{\frac{F_{ij}}{F_{mn}}}$ . Thereby, the heuristic increases capacities of edges closer to the source and decreases capacities of edges closer to the leaves.

The pseudocode is provided in Algorithm 7. The heuristic iterates separately over each path  $p \in P$  in line 1 and then determines the capacities for that path as follows. In lines 2 and 3, the heuristic sets the capacities of all edges in  $p$  to an initial value. This initial value is the minimum capacity such that the constraints for both line sizing and voltage drops are fulfilled. In line 4, the heuristic loops over the depths  $d$  of the edges in  $p$ . In line 5, it selects an edge  $(i, j)$  in the first half of  $p$ , that is,  $d(j) \leq \frac{d(w)}{2}$  where  $d(j)$  is the depth of vertex  $j$ . The corresponding edge  $(m, n)$  further down-

stream in  $p$  is determined in line 6. This edge  $(m, n)$  is as far away from the leaf  $w$  as  $(i, j)$  is from the source 0, i. e.,  $d(m) = d(w) - d(j)$ . In line 7, the heuristic determines the values for the capacities of the two edges  $a_{ij}^{r*}$  and  $a_{mn}^{s*}$  that minimize  $\left| \frac{a_{ij}^r}{a_{mn}^s} - \sqrt{\frac{F_{ij}}{F_{mn}}} \right|$ . While doing so, the capacity  $a_{ij}^r$  of the edge closer to the source is larger than or equal to its initial value  $a_{ij}$ , while the capacity  $a_{mn}^s$  of the edge closer to the leaf is smaller than or equal to its initial value  $a_{mn}$ . Also, the constraints for both line sizing and voltage drops are fulfilled. This optimization problem can be solved via a strategic search by increasing  $a_{ij}^r$  and/or decreasing  $a_{mn}^s$  until  $\left| \frac{a_{ij}^r}{a_{mn}^s} - \sqrt{\frac{F_{ij}}{F_{mn}}} \right|$  does not get any smaller. In line 8, the optimized capacities are stored as candidate solutions. These values are compared with values from earlier iterations because an edge can be part of more than one path. In order to fulfill all constraints for all paths, the maximum value is chosen for edges that have been optimized earlier in line 10. After that, the capacities of all edges have been optimized.

---

**Algorithm 7** Pairwise edge capacity adjustment
 

---

**Input:** Network layout  $\{x_{ij}\}$

- 1: **for** each path  $p$  connecting a leaf  $w$  to the source 0 **do**
  - 2:     **for** each edge  $(i, j) \in p$  **do**
  - 3:          $a_{ij} \leftarrow \min_k \left\{ a_{ij}^k \mid a_{ij}^k \geq \max \left\{ F_{ij}, \frac{1}{Q} \sum_{(i,j) \in p} l_{ij} F_{ij} \right\} \right\}$       $\triangleright$  Set initial value for capacities
  - 4:     **for**  $d \in \{1, \dots, \lfloor d(w)/2 \rfloor\}$  **do**
  - 5:         Select edge  $(i, j) \in p$  with depth  $d(j) = d$       $\triangleright$  Select edge in first half of  $p$
  - 6:         Select edge  $(m, n) \in p$  with  $d(w) - d(m) = d(j)$       $\triangleright$  Select corresponding edge in second half of  $p$
  - 7:         **Compute**  $r, s \leftarrow \arg \min_{r, s \in \{1, \dots, |A|\}} \left| \frac{a_{ij}^r}{a_{mn}^s} - \sqrt{\frac{F_{ij}}{F_{mn}}} \right|$       $\triangleright$  Optimize capacities  
            **s. t.**  $a_{ij}^r \geq a_{ij}, a_{mn}^s \leq a_{mn},$       $\triangleright a_{ij}$  can only be increased,  $a_{mn}$  can only be decreased  
             $a_{mn}^s \geq F_{mn}, \sum_{(v,w) \in p} l_{vw} \frac{F_{vw}}{a_{vw}} \leq Q$       $\triangleright$  Line sizing and voltage drops must be fulfilled
  - 8:          $a_{ij} \leftarrow a_{ij}^{r*}, a_{mn} \leftarrow a_{mn}^{s*}$       $\triangleright$  Set both capacities to optimized values
  - 9:     **for** each edge  $(i, j) \in p$  **do**
  - 10:          $\tilde{a}_{ij} \leftarrow \max\{a_{ij}, \tilde{a}_{ij}\}$       $\triangleright$  Overwrite previous values, if necessary
  - 11: **while** true **do**      $\triangleright$  Post-hoc capacity adjustment
  - 12:     **for** each edge  $(i, j) \in \{\tilde{E} \mid x_{ij} = 1\}$  **do**
  - 13:          $k \leftarrow \{\kappa \mid \tilde{a}_{ij} = a_{ij}^{\kappa}\}$       $\triangleright$  Index lookup
  - 14:         **if**  $k \neq 1$  **then**
  - 15:              $\tilde{a}_{ij} \leftarrow a_{ij}^{k-1}$       $\triangleright$  Decrease capacity
  - 16:         **if**  $a_{ij}^{k-1} \geq F_{ij}$  **and**  $\sum_{(v,w) \in p} l_{vw} \frac{F_{vw}}{\tilde{a}_{vw}} \leq Q$  **for all**  $\{p \mid (i, j) \in p\}$  **then**      $\triangleright$  Check constraints
  - 17:             **continue**
  - 18:             **else**  $\tilde{a}_{ij} \leftarrow a_{ij}^k$       $\triangleright$  Reset capacity, if constraints are violated
  - 19:     **break**
  - 20: **return**  $\{\tilde{a}_{ij}\}$
-

The heuristic now conducts a post-hoc capacity adjustment in lines 11 to 19 for the following reason: some capacities might be larger than needed (i. e., the previous minimization problem has introduced slack capacities), which can be reduced further in lines 3 and 10.

The runtime of this heuristic depends on the network layout. In a starred network, every leaf is directly connected to the source 0. Here the heuristic has a runtime of  $\Theta(N)$  and returns the optimal solution to the capacity optimization problem. For other layouts, runtimes are higher as the number of leaves and the depth of the paths increase with a growing  $N$  (cf. Steele et al., 1987). Then the heuristic has a runtime of at least  $O(N)$ .

The PECA heuristic provides an approximate solution to the capacity optimization problem with theoretical guarantees. To this end, upper bounds for the error can be provided. The formulas provided in Propositions 4 and 5 refer to the maximum error made when optimizing an edge pair in line 7 of the heuristic. The error for the entire network can be calculated by summing over all edge pairs. For this reason, the following notation is introduced. Let  $\Delta_c$  be the cost difference between the optimal cost  $c^*$  for an edge pair and the cost for the same edge pair as determined by the PECA heuristic  $c^{\text{PECA}}$ , i. e.,  $\Delta_c = c^{\text{PECA}} - c^*$ . Let further be  $\Delta_a$  the maximum difference between any two consecutive capacities  $a_{ij}^k, a_{ij}^{k+1} \in A$ , and  $\Delta_l$  the difference in the length of the two edges.

**Proposition 4** *For the network determined by the PECA heuristic, the error in cost for each edge pair is bounded by  $\Delta_c \leq c_m \Delta_a \Delta_l$ .*

*Proof.* Without loss of generality, we assume that the capacities  $a_{ij}^k \in A$  are ordered, with the first element  $a_{ij}^1$  being the smallest. For now, we assume that the capacities in  $A$  are equally spaced, i. e., that  $a_{ij}^{k+1} - a_{ij}^k = \Delta_a$ , for all  $k$ . In case of non-equally spaced values, we define  $\Delta_a = \max_k \{a_{ij}^{k+1} - a_{ij}^k\}$  as the maximum difference between any two subsequent values in  $A$ . Let us consider a pair of edges  $x$  and  $y$  for which

the capacity is subject to optimization using the PECA heuristic. Let  $l_x$  and  $l_y$  be the lengths of these edges.

We assume that the the optimal solution to the capacity optimization problem differs from the one found by the PECA heuristic. Let  $a^x$  and  $a^y$  be the capacities of the optimal solution. Without loss of generality, we assume that  $a^x > a^y$ , and hence,  $x > y$ . Both  $a^x$  and  $a^y$  cannot be identical, since, otherwise the PECA heuristic would have identified them as the optimal solution. Using the cost function in Equation (III-37), the optimal combined cost  $c^*$  for the two edges can be written down. This gives

$$c^* = c_m l_x a^x + c_m l_y a^y . \quad (\text{III-41})$$

Owing to the discrete nature of the CAVLP, the PECA heuristic might determine capacities different from the optimal values  $a^x$  and  $a^y$  due to rounding. One of the edges must then have a capacity larger than in the optimal case, whereas the other must be smaller. Without loss of generality, we can assume that the heuristic has chosen the pair  $a^{x+1}$  and  $a^{y-1}$  as we obtain the same result if the pair  $a^{x-1}$  and  $a^{y+1}$  is chosen. Furthermore, we arrive at the same result if the capacities determined by the PECA heuristic differ from the optimal solution by more than one index.

The cost determined by the PECA heuristic then is

$$c = c_m l_x a^{x+1} + c_m l_y a^{y-1} . \quad (\text{III-42})$$

The difference between this cost and the optimal cost is obtained by subtracting Equation (III-41) from Equation (III-42). This gives

$$\Delta_c = c - c^* = c_m [l_x a^{x+1} + l_y a^{y-1} - l_x a^x - l_y a^y] \quad (\text{III-43})$$

$$= c_m [(a^{x+1} - a^x)l_x - (a^y - a^{y-1})l_y] \quad (\text{III-44})$$

$$= c_m \Delta_a \Delta_l . \square \quad (\text{III-45})$$

As a side observation, Proposition 4 implies that, for networks with approximately equal edge lengths, the error approaches zero (Corollary 2).

**Corollary 2** *If the length difference of an edge pair approaches zero, the cost error approaches zero, i. e.,  $\Delta_c \xrightarrow{\Delta_l \rightarrow 0} 0$ . Therefore, if all edges in the network are of equal length, the error for the entire network approaches zero.*

Below, an upper bound for the error is provided. The formula for this error is independent of the edge lengths. Let  $a_{ij}$  be the selected capacity of the edge closer to the source.

**Proposition 5** *For any edge pair optimized, the relative cost error is  $\frac{\Delta_c}{c^*} \leq 1 / \left( \frac{a_{ij}}{\Delta_a} - 1 \right)$ .*

*Proof.* The same notation as in the proof of Proposition 4 is used. The length and capacity of the edge closer to the source are referred to by the index  $x$  and of the other edge in the edge pair by  $y$ . Let  $a^x$  and  $a^y$  denote the capacities of the optimal solution to the capacity optimization problem. Without loss of generality, it can be assumed that  $l_x = l_y + \Delta_l$ . We obtain

$$\frac{\Delta_c}{c^*} \leq \frac{c_m \Delta_a \Delta_l}{c_m l_x a^x + c_m l_y a^y} = \frac{\Delta_a \Delta_l}{(l_y + \Delta_l) a^x + l_y a^y}. \quad (\text{III-46})$$

We substitute the optimal capacity  $a^x$  in the denominator by the one determined by the PECA heuristic using  $a^x = a^{x-1}$  and obtain

$$\frac{\Delta_c}{c^*} \leq \frac{\Delta_a \Delta_l}{(l_y + \Delta_l) a^{x-1} + l_y a^y}. \quad (\text{III-47})$$

We now analyze the border cases of this expression. In the best case, we obtain

$$\lim_{\Delta_l \rightarrow 0} \frac{\Delta_c}{c^*} = 0. \quad (\text{III-48})$$

In the worst case, the upper bound for the relative cost difference is

$$\lim_{\Delta_l \rightarrow \infty} \frac{\Delta_c}{c^*} \leq \frac{\Delta_a}{a^{x-1}} = \frac{\Delta_a}{a^x - \Delta_a} = \frac{1}{\frac{a_{ij}}{\Delta_a} - 1}. \square \quad (\text{III-49})$$

From Proposition 5, it follows that the error decreases the smaller the difference between  $a_{ij}^{k-1}$  and  $a_{ij}^k$  is.

### 3 Computational Experiments

The developed heuristics are evaluated on a set of simulated network instances with regard to runtime and solution quality. In Section 3.1, the experimental setup is presented. The experiments are conducted for two cases: low and high demand. The results for the low demand case are reported in Section 3.2 and the high demand case are reported in Section 3.3.

#### 3.1 Experimental Setup

The parameter configuration for the numerical experiments is presented in Table III-2. For simplicity, the units for demand, flow and voltage have been dropped. To resemble real-world conditions, the experiments are conducted on network instances of various sizes  $N \in \{5, 10, \dots, 50\}$ . For each  $N$ , 50 instances are generated as follows. Later, the averaged solution quality (as well as the coefficient of variation) is reported. The  $x$ - and  $y$ -locations of the vertices  $(s_x, s_y)$  are sampled from a discrete uniform distribution without replacement. This ensures a realistic setting analogous to electricity distribution networks, where households have different, non-overlapping locations and a certain minimum distance among them. We set  $U_{\text{crit}}$  to an amount in a similar order of magnitude as in reality (e. g., by the partnering electricity power company in the real-world case study). The peak demand per household is set to  $D^{\text{peak}} = 0.01$ . For the coincidence factor, a common function from the literature is

assumed. This function is given by  $\gamma(|I_j|) = \gamma_{\text{lim}} + (1 - \gamma_{\text{lim}}) |I_j|^{-1/2}$  with  $\gamma_{\text{lim}} = 0.1$  (cf. Dickert and Schegner, 2010). This function is monotonically decreasing, convex, and approaches a threshold  $\gamma_{\text{lim}}$  as networks become larger, i. e., ( $|I_j| \rightarrow \infty$ ).

Network size	$x$ -, $y$ -values of vertices	Edge capacities	Voltage drop threshold	Peak demands	Coincidence factor	Costs
$N \in \{5, 10, \dots, 50\}$	$s_x, s_y \sim \text{unif}(0, 50)/10$ ; discrete uniform distribution; sampling without replacement	$a_{ij}^k \sim \{0.1, 0.2, \dots, 1.0\}$	$U_{\text{crit}} = U - 1.0$	$D^{\text{peak}} = 0.01$	$\gamma( I_j ) = 0.1 + (1 - 0.1)  I_j ^{1/2}$	$c_c = c_m = 1$

Table III-2: Parameters for computational experiments.

The evaluation scheme for the heuristics works as follows. The three layout generation heuristics are combined with the five layout improvement heuristics (which includes the two metaheuristics). This results in 15 different combinations. One combination (i. e., starred network + INB heuristic) is excluded from the evaluation, since it is infeasible for the improvement heuristic to change the initial network (a starred network exhibits the highest possible branching). The solution quality of the initial network layout without any improvement heuristic is also reported.

Additionally, two approaches to find the exact solution to the CAVLP are tried. The first is by implementing the linearized version of the CAVLP and solving it using the Gurobi Optimizer 7.5.2 MIP solver (as described in Section 1.5). The second is by complete enumeration of all possible layouts and capacities. For both approaches, the optimal solution is reported, or, in case an optimal solution could not be found within the given time limit, the best solution obtained is reported.

An upper and lower bound for the exact solution is reported. The bounds are determined by constantly over- or underestimating the demand using a uniform coincidence factor for all edges in the network. This enables an implementation of a simplified version of the CAVLP with the Gurobi Optimizer 7.5.2 MIP solver. For the upper bound, a coincidence factor of  $\gamma \equiv 1$  is chosen. This corresponds to a scenario of fully coinciding peak loads. For the lower bound, a coincidence factor of  $\gamma \equiv \gamma(N - 1)$  is applied as an overall discount factor to all loads. This corresponds to the maximum achievable discount for a network of size  $N$ . It is still not possible to solve the simplified problem specification in an exact manner for larger instances

( $N > 20$ ). In this case, the best solution from Gurobi is reported that was reached within the time limit.

For all heuristic solutions, the capacity optimization is carried out using the PECA heuristic. In Section 6.2, a sensitivity analysis is presented where the PECA heuristic is compared against alternative approaches for capacity optimization. Here the results show that the solution quality is on par with that from an exact solver, yet has a substantially lower runtime.

The following parameters are used for the improvement heuristics: The random improvement heuristic uses  $s_{\max} = 3N$ . The increased and decreased branching heuristics use the parameter  $d_{\max} = 4$ . The VNS algorithm uses  $d_{\max} = 3$ ,  $s_{\text{VNS}} = 5$ , and  $s_{\max} = N$  for the local search. The Tabu Search algorithm uses  $s_{\text{tabu}} = 10N$ . For instances with  $N \leq 20$ , the length of the tabu list is set to 5; for all other instances, it is set to 10.

The heuristics are implemented in Python 3.5. All computational experiments are conducted in parallel on 4 multi-core Intel Xeon E5-2630 v4 CPUs at 2.2 GHz and 8 GB of RAM. 16 computations are running in parallel at any given time. This process does not impair the individual runtimes. The runtime limit for each instance of size  $N$  is set to 24 hours for 50 instances. Note that experiments with larger runtime limits were also conducted; yet, due to the problem complexity, exact solutions are prevented.

## 3.2 Results for High Demand Case

### 3.2.1 Solution Quality

Table III-3 displays the performance of the heuristics in terms of cost. First, the performance of the heuristics for the generation of the initial layout is evaluated. For this purpose, the cost for the initial layouts without a subsequent improvement heuristic are compared (given in the first row for each layout generating heuristic). For small networks of up to fifteen loads, the MST performs best, followed closely by the greedy

network construction heuristic. For networks with 20 or more loads, the greedy network construction heuristic slightly outperforms the MST. This is explained by the fact that, as the networks grow larger, the electricity flows close to the source increase. This observation is utilized by the greedy network construction (as opposed to the MST), as it creates layouts that are more centered around the source. The MST still appears to provide a better basis when being combined with an improvement heuristic. The reason is that the MST algorithm provides an initial network layout that has a relatively low branching. The branching can subsequently be increased, which is beneficial for the improvement heuristics. The starred network algorithm performs considerably worse. This rather poor performance of the starred network algorithm is to be expected, since the algorithm is designed for high demand scenarios.

Initial Layout	Layout Improvement	$N$	5	10	15	20	25	30	35	40	45	50
MST	No heuristic		7.37 (0.23)	12.28 (0.14)	15.66 (0.12)	18.92 (0.10)	21.26 (0.10)	24.03 (0.11)	25.92 (0.11)	28.68 (0.11)	30.64 (0.10)	32.02 (0.09)
	Randomized Reconfiguration		7.35 (0.23)	12.16 (0.14)	15.49 (0.11)	18.54 (0.10)	20.65 (0.08)	23.13 (0.08)	24.84 (0.08)	27.19 (0.07)	28.89 (0.07)	30.72 (0.06)
	INB		7.35 (0.23)	12.17 (0.14)	15.48 (0.11)	18.56 (0.10)	20.65 (0.09)	23.02 (0.08)	24.92 (0.09)	27.39 (0.07)	28.95 (0.07)	30.70 (0.07)
	DNB		7.36 (0.23)	12.21 (0.14)	15.59 (0.12)	18.71 (0.10)	20.87 (0.09)	23.48 (0.10)	25.33 (0.09)	27.81 (0.09)	29.72 (0.08)	31.04 (0.07)
	VNS		7.35 (0.23)	12.14 (0.14)	15.43 (0.11)	18.42 (0.09)	20.53 (0.08)	22.87 (0.07)	24.75 (0.08)	27.06 (0.07)	28.66 (0.07)	30.48 (0.06)
	Tabu Search		7.35 (0.23)	12.14 (0.14)	15.41 (0.11)	18.39 (0.09)	20.45 (0.08)	22.72 (0.07)	24.55 (0.08)	26.77 (0.06)	28.42 (0.07)	30.13 (0.06)
Starred Network Algorithm	No heuristic		11.23 (0.31)	26.40 (0.21)	40.39 (0.19)	56.25 (0.18)	68.67 (0.18)	82.27 (0.16)	99.99 (0.19)	117.64 (0.20)	122.26 (0.16)	140.97 (0.19)
	Randomized Reconfiguration		7.44 (0.23)	14.75 (0.19)	21.41 (0.12)	29.82 (0.12)	35.71 (0.14)	43.13 (0.10)	51.14 (0.10)	59.84 (0.11)	66.16 (0.09)	73.78 (0.09)
	DNB		7.62 (0.24)	12.55 (0.16)	15.95 (0.12)	19.17 (0.10)	21.30 (0.10)	23.56 (0.08)	25.51 (0.08)	27.86 (0.07)	29.72 (0.07)	32.05 (0.07)
	VNS		7.35 (0.23)	13.15 (0.14)	18.46 (0.11)	24.39 (0.11)	28.82 (0.12)	35.15 (0.10)	41.15 (0.11)	47.92 (0.10)	52.52 (0.09)	59.33 (0.07)
	Tabu Search		7.35 (0.23)	12.30 (0.14)	16.61 (0.11)	20.99 (0.12)	25.08 (0.11)	28.92 (0.10)	33.35 (0.10)	38.16 (0.08)	41.41 (0.09)	45.77 (0.08)
Greedy Network Construction	No heuristic		7.42 (0.23)	12.43 (0.15)	15.93 (0.11)	18.86 (0.09)	21.20 (0.08)	23.63 (0.08)	25.66 (0.08)	28.12 (0.08)	29.64 (0.07)	31.44 (0.06)
	Randomized Reconfiguration		7.35 (0.23)	12.23 (0.14)	15.69 (0.11)	18.58 (0.09)	20.87 (0.07)	23.30 (0.07)	25.34 (0.08)	27.62 (0.08)	29.30 (0.07)	30.95 (0.06)
	INB		7.40 (0.23)	12.35 (0.14)	15.81 (0.11)	18.79 (0.09)	21.08 (0.08)	23.44 (0.07)	25.42 (0.08)	27.71 (0.07)	29.38 (0.06)	31.11 (0.06)
	DNB		7.36 (0.23)	12.19 (0.14)	15.66 (0.11)	18.58 (0.09)	20.82 (0.08)	23.15 (0.08)	25.16 (0.08)	27.53 (0.07)	29.14 (0.07)	30.99 (0.06)
	VNS		7.35 (0.23)	12.15 (0.14)	15.51 (0.11)	18.50 (0.09)	20.77 (0.08)	23.14 (0.07)	25.12 (0.08)	27.58 (0.07)	29.19 (0.07)	30.95 (0.06)
Exact solution	Tabu Search		7.35 (0.23)	12.14 (0.14)	15.48 (0.11)	18.42 (0.09)	20.47 (0.08)	22.78 (0.07)	24.89 (0.08)	27.12 (0.07)	28.81 (0.06)	30.46 (0.06)
	Gurobi MIP solver		7.34 (0.23)	12.26 (0.14) <sup>†</sup>	22.89 (0.30) <sup>‡</sup>	— <sup>†</sup>	— <sup>†</sup>	— <sup>†</sup>				
Bounds	Complete enumeration		7.34 (0.23)	15.66 (0.13) <sup>†</sup>	27.72 (0.10) <sup>†</sup>	42.91 (0.08) <sup>†</sup>	58.54 (0.09) <sup>†</sup>	77.10 (0.08) <sup>†</sup>	96.09 (0.06) <sup>†</sup>	119.22 (0.08) <sup>†</sup>	142.89 (0.08) <sup>†</sup>	170.9 (0.06) <sup>†</sup>
	Upper bound		7.49 (0.24)	12.66 (0.15)	16.40 (0.12) <sup>†</sup>	20.16 (0.11) <sup>†</sup>	23.27 (0.12) <sup>†</sup>	26.18 (0.09) <sup>†</sup>	29.57 (0.15) <sup>†</sup>	33.07 (0.15) <sup>†</sup>	35.01 (0.14) <sup>†</sup>	39.26 (0.20) <sup>†</sup>

<sup>†</sup> Runtime exceeded time limit, reported is the best solution within the time limit; <sup>‡</sup> No viable solution found within time limit

Table III-3: Comparison of network cost for various number of loads  $N$ . Reported is the cost averaged across 50 instances with the coefficient of variation in parenthesis. The cells are shaded based on average cost. Threshold for timeout: 24 hours per all 50 instances.

Second, let's look at the results of combining the heuristics for generating the network layout and the layout improvement heuristics. Overall, the best results are achieved using a combination of MST algorithm and Tabu Search, followed closely by a combination of MST and VNS, as well as greedy network construction and Tabu Search. These three combinations of heuristics yield very similar results which only differ in between 0.0% and 1.4% across network sizes. Moreover, there are several other combinations of heuristics that perform only slightly worse, such as greedy con-

struction and VNS, MST and INB, MST and DNB, MST and random reconfiguration, as well as greedy network construction and random reconfiguration. In general, the ranking and the performance differences across all combinations of heuristics remain consistent, which is an indication of the effectiveness of the solution approach.

Third, the heuristics for layout improvements are evaluated. Overall, the meta-heuristics (VNS and Tabu Search) yield very similar results but outperform the other local improvement heuristics for most instance sizes and layout generation heuristics. The Tabu Search algorithm achieves slightly better results than VNS, particularly for larger instances. Among the other improvement heuristics, the INB heuristic is primarily effective for initial layouts that have a low branching (such as those generated by MST). DNB heuristic is effective when operating on initial network layouts with a higher branching factor. It is the only improvement heuristic that performs consistently well in combination with the starred network. The randomized network reconfiguration performs reasonably well for most initial layouts (with the exception of the starred network layout).

Fourth, regarding the exact results and bounds, the following three observations can be made. (1) Exact solutions can only be obtained for small networks with  $N = 5$  due to NP-hardness. For  $N = 5$ , the majority of heuristics find the optimal solution for most network instances. The slightly higher average cost reported in comparison to the exact solution approaches results from 4–5 network instances where the heuristic solutions are marginally inferior. For  $N = 10$ , the MIP solver reaches the timeout for about half the instances. For  $N = 15$ , the timeout is reached for every instance and the average network cost determined by Gurobi is 49% more expensive than the best solution found by the heuristics (while runtimes are more than 500-times higher). For  $N > 15$ , the solver is unable to determine viable solutions. For  $N \geq 10$ , the networks obtained by complete enumeration within the time limit exhibit much higher costs than the ones obtained by any of the presented solution methods (except the starred network without layout improvement). Note that experiments with much higher runtimes (several days) were conducted for the solver, yet still no viable solutions

were obtained for larger instances. (2) The heuristics remain considerably close to the lower bound. For small instances ( $N \leq 20$ ), the best performing heuristics remain within 2.8% of the lower bound. For large instances ( $N > 20$ ), the distance increases only moderately to 4.7%. This underlines the effectiveness of the heuristics. (3) The majority of the heuristics reach considerably lower cost than the upper bound. The advantage is consistent and increases with network sizes in both absolute and relative terms. This shows the advantage of taking into account varying coincidence factors.

In summary, several combinations of heuristics perform well. The analysis of the optimal solution and the bounds furthermore points towards the overall effectiveness of the solution heuristics. Finally, a low coefficient of variation is attained by all heuristics, thereby indicating that, independent of the individual problem setting, the different methods are stable in their performance.

### 3.2.2 Runtime

Table III-5 lists the runtimes of the different methods. As expected, the average runtime increases with the instance size. For instances larger than  $N = 10$ , an exact solution becomes intractable. For 15 vertices, not even one single optimal solution could be found. It is reiterated that larger runtime limits of several days have been tested, yet this did not help in finding solutions for larger instances due to the complexity of the problem. Hence, exact solutions for real-world-sized problems are precluded and, instead, heuristics must be used.

Overall, the heuristics achieve runtimes that, for real-world applications, are sufficient. In general, the metaheuristics have longer runtimes than the local improvement heuristics (but achieve lower costs). All methods reveal a low coefficient of variation, which highlights that there is fairly little variance in the runtime.

Initial Layout	Layout Improvement	$N$	5	10	15	20	25	30	35	40	45	50
MST	No heuristic		0.00 (0.38)	0.00 (0.41)	0.01 (0.53)	0.01 (0.55)	0.03 (0.69)	0.07 (0.66)	0.11 (0.86)	0.18 (0.84)	0.26 (0.56)	0.36 (0.57)
	Randomized Reconfiguration		0.01 (0.24)	0.19 (0.32)	0.98 (0.37)	4.21 (0.35)	11.28 (0.39)	31.24 (0.44)	64.58 (0.37)	108.87 (0.46)	194.63 (0.35)	315.37 (0.41)
	INB		0.00 (0.65)	0.07 (0.61)	0.71 (0.57)	3.24 (0.56)	10.24 (0.49)	29.58 (0.57)	68.01 (0.75)	124.18 (0.55)	266.97 (0.52)	533.01 (1.04)
	DNB		0.00 (0.23)	0.15 (0.27)	1.01 (0.24)	4.34 (0.37)	12.09 (0.29)	29.44 (0.38)	55.01 (0.45)	102.77 (0.42)	186.64 (0.47)	273.21 (0.47)
	VNS		0.19 (0.18)	1.64 (0.28)	7.13 (0.32)	25.10 (0.32)	63.65 (0.34)	152.17 (0.30)	306.83 (0.35)	536.97 (0.40)	893.21 (0.35)	1504.26 (0.44)
Starred Network Algorithm	Tabu Search		0.05 (0.20)	0.60 (0.28)	2.98 (0.32)	12.76 (0.37)	34.10 (0.36)	82.22 (0.37)	195.59 (0.42)	343.44 (0.43)	536.97 (0.40)	899.84 (0.40)
	No heuristic		0.00 (0.11)	0.00 (0.07)	0.00 (0.06)	0.00 (0.06)	0.00 (0.04)	0.00 (0.04)	0.00 (0.04)	0.00 (0.03)	0.00 (0.13)	0.00 (0.05)
	Randomized Reconfiguration		0.01 (0.21)	0.11 (0.34)	0.38 (0.44)	1.14 (0.58)	1.96 (0.53)	3.66 (0.64)	8.08 (0.78)	12.55 (0.69)	13.35 (0.64)	22.13 (0.72)
	DNB		0.01 (0.31)	0.23 (0.31)	1.49 (0.28)	6.33 (0.28)	17.54 (0.25)	38.24 (0.25)	89.05 (0.34)	156.60 (0.36)	275.76 (0.34)	373.82 (0.34)
	VNS		0.12 (0.19)	0.39 (0.30)	1.29 (0.36)	3.48 (0.44)	7.00 (0.48)	13.11 (0.47)	26.63 (0.58)	44.31 (0.58)	52.07 (0.53)	81.76 (0.55)
Greedy Network Construction	Tabu Search		0.04 (0.20)	0.51 (0.27)	2.35 (0.33)	7.74 (0.46)	17.97 (0.40)	34.58 (0.44)	76.57 (0.52)	117.05 (0.45)	162.48 (0.57)	263.42 (0.46)
	No heuristic		0.00 (0.30)	0.00 (0.43)	0.00 (0.46)	0.01 (0.44)	0.02 (0.40)	0.03 (0.50)	0.05 (0.45)	0.08 (0.63)	0.09 (0.45)	0.13 (0.40)
	Randomized Reconfiguration		0.01 (0.25)	0.18 (0.30)	0.86 (0.37)	3.60 (0.40)	10.13 (0.44)	22.1 (0.42)	47.7 (0.42)	85.8 (0.46)	151.34 (0.39)	221.86 (0.39)
	INB		0.00 (0.65)	0.06 (0.71)	0.63 (0.68)	2.94 (0.54)	9.76 (0.49)	26.41 (0.62)	58.20 (0.50)	101.83 (0.45)	209.95 (0.62)	361.34 (0.66)
	DNB		0.00 (0.24)	0.16 (0.25)	1.08 (0.27)	4.82 (0.31)	13.79 (0.23)	31.96 (0.33)	59.24 (0.37)	106.47 (0.33)	179.81 (0.34)	251.36 (0.34)
Exact solution	VNS		0.18 (0.17)	1.45 (0.27)	5.98 (0.29)	21.92 (0.41)	55.83 (0.40)	111.55 (0.39)	253.55 (0.37)	437.15 (0.41)	688.96 (0.38)	1160.11 (0.40)
	Tabu Search		0.05 (0.19)	0.59 (0.27)	2.80 (0.32)	11.77 (0.41)	31.64 (0.45)	71.98 (0.42)	155.42 (0.43)	268.49 (0.42)	460.96 (0.36)	729.92 (0.43)
	Gurobi MIP solver		2.74 (1.06)	988.23 (0.73) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>
	Complete enumeration		75.60 (0.05)	1728.00 (0.00) <sup>†</sup>								

<sup>†</sup> Calculation timed-out

Table III-4: Comparison of runtimes for various number of loads  $N$ . Reported is the runtime in seconds averaged across 50 instances with the coefficient of variation in parenthesis. The cells are shaded based on average runtime. Threshold for timeout: 24 hours per all 50 instances.

### 3.3 Results for High Demand Case

The evaluation of the solution methods using a high demand case is done below. The same experimental setup and parameters as before are used. The only difference is that the peak demand is now set to  $D^{\text{peak}} = 0.02$ , i. e., twice as large as in the low demand case.

#### 3.3.1 Solution Quality

The high demand case is characterized by considerably larger electricity flows and thus larger voltage drops. This has a direct implication: the heuristics cannot always find a feasible solution. That is, the heuristics for generating the network layout provide a layout for which the constraint for the voltage drops cannot be fulfilled even when all capacities are set to the maximum possible value. Section 3.3.1 reports the number of infeasible networks that were created by the different heuristics. Note that such infeasible solutions can evidently be prevented when using any combination of the starred network algorithm. This is ensured by the theoretical properties of the heuristic (cf. Remark 3). The greedy network construction returns a few solutions that are infeasible but only for larger networks and when not paired with an improvement heuristic. In contrast, the MST produces infeasible solutions even for instances as small as 25 vertices. For networks larger than 35 vertices, about half

of the MST layouts are infeasible, suggesting that alternative optimization methods should be preferred in a high demand case.

Initial Layout	Layout Improvement	N	5	10	15	20	25	30	35	40	45	50
<b>MST</b>	No heuristic	0	0	0	0	0	3	12	13	22	27	24
	Randomized Reconfiguration	0	0	0	0	0	0	0	6	11	13	16
	INB	0	0	0	0	0	0	0	0	3	4	3
	DNB	0	0	0	0	0	0	1	1	8	9	9
	VNS	0	0	0	0	0	0	0	1	4	4	11
	Tabu Search	0	0	0	0	0	0	0	5	13	12	17
<b>Starred Network Algorithm</b>	No heuristic	0	0	0	0	0	0	0	0	0	0	0
	Randomized Reconfiguration	0	0	0	0	0	0	0	0	0	0	0
	DNB	0	0	0	0	0	0	0	0	0	0	0
	VNS	0	0	0	0	0	0	0	0	0	0	0
	Tabu Search	0	0	0	0	0	0	0	0	0	0	0
<b>Greedy Network Construction</b>	No heuristic	0	0	0	0	0	0	0	0	4	0	0
	Randomized Reconfiguration	0	0	0	0	0	0	0	0	0	0	0
	INB	0	0	0	0	0	0	0	0	0	0	0
	DNB	0	0	0	0	0	0	0	0	0	0	0
	VNS	0	0	0	0	0	0	0	0	0	0	0
	Tabu Search	0	0	0	0	0	0	0	0	0	0	0
<b>Exact solution</b>	Gurobi MIP solver	0	0	—†	—†	—†	—†	—†	—†	—†	—†	—†
	Complete Enumeration	0	0	0	0	0	0	0	0	0	0	0

† Calculation timed-out

Table III-5: Comparison of infeasible networks produced in high demand case for various number of loads  $N$ . In total, 50 randomly generated instances were produced for each  $N$ . Threshold for timeout: 24 hours per all 50 instances.

The cost performance of the heuristics is shown in Table III-6. For comparability, only those instances are included whereby a feasible solution was found by all heuristics. Overall, costs are higher than in the low demand case because the networks need to have larger capacities and higher branching. For  $N = 5$ , most heuristics yield solutions that register costs similar to those from the exact solution. This points towards the effectiveness of the proposed heuristics. For  $N > 5$ , the best solution obtained by complete enumeration within the time limit is reported. Similar to the low demand case, these networks exhibit much higher costs than the ones obtained by any of the proposed solution methods (except the starred network without layout improvement).

The metaheuristics (VNS and Tabu Search) outperform the local improvement heuristics, with Tabu Search performing best, particularly for larger instances. The INB heuristics registers a superior performance among the local improvement heuris-

tics. The lowest cost is still obtained with combinations starting with the MST. However, keep in mind the large number of infeasible networks that are excluded from this analysis. For all heuristics, the performance is fairly stable as demonstrated by a low coefficient of variation. Furthermore, it can be observed that the greedy network construction is providing the best initial layout (without applying an improvement heuristic).

Regarding the the exact results and bounds, similar observations than in the low demand case can be made. (1) Exact solutions can only be obtained for the smallest instances with  $N = 5$  due to NP-hardness. For  $N = 5$ , the majority of heuristics find network layouts with similar costs to the optimal solution. In the high demand case, Gurobi is only able to determine viable solutions for networks with  $N \leq 10$  (as opposed to  $N \leq 15$  in the low demand case). For  $N > 5$ , the networks determined by complete enumeration or Gurobi within the time limit exhibit much higher costs than the ones obtained by any of the solution methods (except the starred network without layout improvement). (2) The heuristics remain considerably close to the lower bound. The best performing heuristics remain within 6.6 % of the lower bound. This gap is slightly larger than in the low demand case. This is to be expected, because, for the lower bound, all demands are consistently underestimated. This has a larger effect in the high demand case. (3) The majority of the heuristics reach considerably lower costs than the upper bound. The advantage is consistent and increases with network sizes in both absolute and relative terms.

### 3.3.2 Runtime

Table III-7 reports the runtimes in the high demand case. Analogous to the above cost analysis, the figures shown are only averages over those instances for which all heuristics can find a feasible network. First of all, for the INB and DNB heuristics, the computation takes—on average—longer than in the low demand case presented before. This is explained by the fact that networks in the high demand case tend to be characterized by a higher branching. Second, for the randomized network recon-

Initial Layout	Layout Improvement	N	5	10	15	20	25	30	35	40	45	50
MST	No heuristic	7.74 (0.25)	13.44 (0.17)	17.36 (0.15)	21.55 (0.13)	23.90 (0.11)	26.43 (0.11)	28.58 (0.10)	31.06 (0.10)	33.09 (0.08)	35.02 (0.08)	
	Randomized Reconfiguration	7.69 (0.25)	13.09 (0.15)	16.80 (0.12)	20.39 (0.10)	22.64 (0.09)	25.18 (0.08)	27.50 (0.09)	29.27 (0.08)	31.49 (0.07)	33.75 (0.07)	
	INB	7.70 (0.25)	13.04 (0.15)	16.79 (0.12)	20.36 (0.11)	22.58 (0.09)	25.05 (0.08)	27.12 (0.09)	29.22 (0.08)	30.96 (0.08)	33.13 (0.06)	
	DNB	7.72 (0.25)	13.15 (0.15)	17.12 (0.15)	20.89 (0.11)	23.04 (0.09)	25.72 (0.10)	27.74 (0.10)	29.71 (0.08)	31.69 (0.09)	33.87 (0.07)	
	VNS	7.69 (0.25)	12.99 (0.15)	16.71 (0.12)	20.13 (0.10)	22.46 (0.09)	24.81 (0.08)	27.23 (0.09)	29.36 (0.07)	31.21 (0.07)	33.47 (0.07)	
Starred Network Algorithm	Tabu Search	7.69 (0.25)	12.96 (0.15)	16.70 (0.12)	20.02 (0.10)	22.28 (0.09)	24.65 (0.08)	26.80 (0.08)	28.89 (0.08)	30.64 (0.08)	32.98 (0.06)	
	No heuristic	11.27 (0.31)	26.57 (0.22)	40.55 (0.19)	56.61 (0.18)	67.75 (0.18)	79.90 (0.17)	96.08 (0.19)	106.87 (0.18)	112.98 (0.15)	129.58 (0.17)	
	Randomized Reconfiguration	7.82 (0.26)	15.46 (0.20)	23.52 (0.15)	31.83 (0.12)	38.04 (0.14)	45.91 (0.12)	54.18 (0.13)	62.13 (0.11)	68.45 (0.08)	76.55 (0.09)	
	DNB	7.93 (0.24)	13.30 (0.16)	17.09 (0.12)	20.68 (0.10)	23.10 (0.09)	25.48 (0.09)	28.00 (0.09)	29.65 (0.07)	31.97 (0.08)	34.07 (0.07)	
	VNS	7.69 (0.25)	13.77 (0.15)	19.82 (0.15)	26.33 (0.15)	30.65 (0.11)	38.19 (0.11)	45.33 (0.14)	51.65 (0.13)	56.96 (0.10)	62.55 (0.09)	
Greedy Network Construction	Tabu Search	7.69 (0.25)	13.10 (0.15)	17.89 (0.12)	22.64 (0.11)	26.61 (0.12)	31.84 (0.12)	36.63 (0.09)	41.79 (0.12)	45.23 (0.12)	50.35 (0.11)	
	No heuristic	7.78 (0.25)	13.38 (0.16)	17.38 (0.13)	20.77 (0.10)	23.13 (0.09)	25.67 (0.08)	28.14 (0.09)	30.12 (0.09)	32.27 (0.09)	34.44 (0.08)	
	Randomized Reconfiguration	7.69 (0.25)	13.05 (0.15)	16.98 (0.12)	20.37 (0.10)	22.63 (0.08)	25.19 (0.08)	27.54 (0.09)	29.52 (0.07)	31.64 (0.08)	33.77 (0.07)	
	INB	7.74 (0.25)	13.17 (0.15)	17.01 (0.12)	20.44 (0.10)	22.77 (0.09)	25.20 (0.08)	27.48 (0.09)	29.59 (0.08)	31.37 (0.08)	33.44 (0.06)	
	DNB	7.72 (0.25)	13.07 (0.15)	16.98 (0.13)	20.29 (0.10)	22.66 (0.09)	24.95 (0.08)	27.50 (0.09)	29.48 (0.09)	31.50 (0.09)	33.60 (0.07)	
Exact solution	VNS	7.69 (0.25)	13.00 (0.15)	16.75 (0.12)	20.15 (0.10)	22.50 (0.09)	25.02 (0.07)	27.38 (0.09)	29.50 (0.08)	31.48 (0.09)	33.49 (0.06)	
	Tabu Search	7.69 (0.25)	12.97 (0.15)	16.67 (0.12)	20.04 (0.10)	22.26 (0.08)	24.72 (0.07)	27.05 (0.09)	28.92 (0.07)	30.97 (0.08)	32.94 (0.06)	
	Gurobi MIP solver	7.67 (0.25)	15.00 (0.31) <sup>†</sup>	— <sup>‡</sup>	— <sup>‡</sup>	— <sup>‡</sup>	— <sup>‡</sup>	— <sup>‡</sup>				
Bounds	Complete enumeration	7.67 (0.25)	16.46 (0.15) <sup>†</sup>	29.76 (0.11) <sup>†</sup>	48.52 (0.10) <sup>†</sup>	67.14 (0.11) <sup>†</sup>	91.62 (0.10) <sup>†</sup>	123.00 (0.15) <sup>†</sup>	186.80 (0.20) <sup>†</sup>	202.26 (0.16) <sup>†</sup>	234.93 (0.16) <sup>†</sup>	
	Lower bound	7.55 (0.24)	12.47 (0.14)	15.76 (0.11) <sup>†</sup>	18.78 (0.10) <sup>†</sup>	21.01 (0.09) <sup>†</sup>	23.40 (0.08) <sup>†</sup>	25.49 (0.09) <sup>†</sup>	28.16 (0.10) <sup>†</sup>	29.82 (0.09) <sup>†</sup>	31.77 (0.10) <sup>†</sup>	
	Upper bound	7.94 (0.25)	13.96 (0.16)	18.77 (0.14) <sup>†</sup>	24.06 (0.14) <sup>†</sup>	28.08 (0.17) <sup>†</sup>	31.65 (0.14) <sup>†</sup>	38.86 (0.28) <sup>†</sup>	44.60 (0.25) <sup>†</sup>	48.41 (0.26) <sup>†</sup>	58.87 (0.41) <sup>†</sup>	

<sup>†</sup> Runtime exceeded time limit, reported is the best solution within the time limit; <sup>‡</sup> No viable solution found within time limit

Table III-6: Comparison of network cost for various number of loads  $N$  in high demand case. Figures shown are only averages over those instances whereby a feasible network could be found by every combination. Reported is the average cost with the coefficient of variation in parenthesis. The cells are shaded based on average cost. Threshold for timeout: 24 hours per all 50 instances.

figuration, VNS and Tabu Search the computation is—on average—faster than in the other case. This can be explained easily. All of these methods at some point form a cycle within the network. They delete the most cost-efficient edge within that cycle by looping through all edges of the cycle. As the high demand networks exhibit higher branching, the branches themselves are shorter. This translates into shorter cycles and therefore shorter calculation time. Third, the coefficient of variation is consistently low, thus pointing towards little variability in the runtime.

Initial Layout	Layout Improvement	N	5	10	15	20	25	30	35	40	45	50
MST	No heuristic	0.00 (0.35)	0.00 (0.40)	0.01 (0.64)	0.02 (0.60)	0.04 (0.43)	0.07 (0.49)	0.10 (0.34)	0.16 (0.55)	0.23 (0.37)	0.36 (0.31)	
	Randomized Reconfiguration	0.02 (0.21)	0.32 (0.23)	1.60 (0.32)	6.26 (0.37)	14.57 (0.29)	31.79 (0.29)	55.42 (0.28)	79.48 (0.24)	146.31 (0.27)	229.77 (0.23)	
	INB	0.00 (0.75)	0.10 (0.67)	1.10 (0.51)	5.26 (0.62)	17.39 (0.52)	45.24 (0.54)	94.52 (0.48)	193.94 (0.40)	333.40 (0.40)	626.62 (0.60)	
	DNB	0.01 (0.23)	0.22 (0.27)	1.48 (0.24)	5.46 (0.40)	14.12 (0.33)	25.82 (0.44)	44.34 (0.53)	86.37 (0.45)	135.60 (0.43)	186.34 (0.45)	
	VNS	0.25 (0.19)	2.08 (0.28)	9.08 (0.30)	28.29 (0.43)	66.62 (0.31)	114.81 (0.26)	206.67 (0.31)	330.75 (0.35)	641.51 (0.32)	885.61 (0.25)	
Starred Network Algorithm	Tabu Search	0.06 (0.22)	0.83 (0.27)	4.20 (0.31)	15.9 (0.39)	36.69 (0.35)	76.81 (0.29)	145.17 (0.26)	232.98 (0.29)	414.65 (0.42)	653.47 (0.24)	
	No heuristic	0.00 (0.18)	0.00 (0.13)	0.00 (0.10)	0.00 (0.20)	0.00 (0.13)	0.00 (0.14)	0.00 (0.22)	0.00 (0.23)	0.00 (0.22)	0.00 (0.17)	
	Randomized Reconfiguration	0.02 (0.23)	0.17 (0.39)	0.60 (0.39)	1.51 (0.46)	2.98 (0.53)	5.06 (0.42)	8.98 (0.56)	11.68 (0.62)	15.06 (0.66)	23.19 (0.55)	
	DNB	0.01 (0.29)	0.34 (0.34)	2.24 (0.34)	9.31 (0.22)	24.80 (0.26)	52.9 (0.24)	100.16 (0.23)	178.02 (0.21)	314.63 (0.19)	517.51 (0.24)	
	VNS	0.15 (0.20)	0.50 (0.29)	1.73 (0.40)	4.07 (0.37)	8.38 (0.38)	12.81 (0.38)	22.09 (0.39)	33.50 (0.50)	44.57 (0.39)	69.09 (0.41)	
Greedy Network Construction	Tabu Search	0.06 (0.20)	0.68 (0.27)	3.00 (0.37)	9.74 (0.40)	19.23 (0.38)	35.85 (0.35)	60.76 (0.35)	92.88 (0.33)	128.68 (0.33)	198.55 (0.34)	
	No heuristic	0.00 (1.75)	0.00 (0.41)	0.01 (0.52)	0.01 (0.52)	0.02 (0.41)	0.04 (0.48)	0.06 (0.46)	0.08 (0.64)	0.15 (0.55)	0.20 (0.42)	
	Randomized Reconfiguration	0.02 (0.22)	0.30 (0.27)	1.59 (0.34)	5.59 (0.41)	14.36 (0.34)	26.75 (0.30)	49.77 (0.27)	67.36 (0.24)	124.06 (0.29)	193.33 (0.26)	
	INB	0.00 (0.64)	0.09 (0.64)	0.98 (0.60)	4.84 (0.52)	17.47 (0.51)	46.24 (0.57)	101.68 (0.42)	193.21 (0.42)	358.64 (0.45)	631.04 (0.56)	
	DNB	0.01 (0.25)	0.23 (0.21)	1.62 (0.32)	6.61 (0.35)	17.65 (0.29)	34.12 (0.28)	55.75 (0.39)	93.86 (0.34)	185.67 (0.41)	253.02 (0.36)	
Exact solution	VNS	0.24 (0.19)	1.98 (0.25)	8.27 (0.29)	27.80 (0.39)	62.77 (0.35)	115.64 (0.31)	204.51 (0.34)	302.39 (0.26)	547.09 (0.24)	788.34 (0.27)	
	Tabu Search	0.06 (0.19)	0.82 (0.24)	4.06 (0.34)	14.79 (0.41)	35.64 (0.34)	70.26 (0.33)	128.41 (0.25)	207.72 (0.25)	352.57 (0.23)	557.50 (0.27)	
	Gurobi MIP solver	6.24 (1.41)	1548.93 (0.28) <sup>†</sup>	1728.00 (0.00) <sup>†</sup>								
	Complete enumeration	69.91 (0.07)	1728.00 (0.00) <sup>†</sup>									

<sup>†</sup> Calculation timed-out

Table III-7: Comparison of runtimes for various number of loads  $N$  in high demand case. Figures shown are only averages over those instances whereby a feasible network could be found by every combination. Reported is the average cost with the coefficient of variation in parenthesis. The cells are shaded based on average runtime. Threshold for timeout: 24 hours per all 50 instances.

## 4 Real-World Experiments

The applicability of the proposed solution methods are demonstrated based on a set of low voltage distribution networks from a Swiss distribution network operator. These network instances are used to compare the costs from conventional network design in practice to the costs determined by the proposed solution methods. This quantifies the potential monetary benefit of the proposed methods under real-world conditions.

### 4.1 Experimental Setup

The real-world experiments are based on a sample of 74 low voltage distribution networks in Switzerland. Each network entails one transformer. The number of loads per network in the sample ranges from 12 to 68, with an average of 36.7 loads and a median of 32 loads. These loads correspond to the vertices of the network. The costs for the networks range from CHF 33,500 for the cheapest network to CHF 1.7 million for the most expensive. The average network cost is CHF 251,400. The networks sum up to a combined value of CHF 18.6 million. This cost only includes material and construction and excludes planning and overhead costs. On average, each network covers an area of 35.1 ha, i. e., 0.351 km<sup>2</sup>, with a median size of 20.6 ha per network.

The networks are grouped by the number of loads  $N$  contained in each network. The reason for this grouping is that the performance of the solution methods varies according to network size, and, this way, the results can be compared to the computational experiments from above. The groups of networks range from  $N \in [10, 19]$  to  $N \in [60, 69]$ . The number of networks in each group is reported alongside the results. The largest group contains 19 networks and is given by  $N \in [20, 29]$ .

The network data (i. e., longitude and latitude of loads, network layout, and capacities) have been extracted from a geographic information system. All distances between locations are computed using the Euclidean distance (L2 norm). Some of the locations in the original data set belong to components with zero energy demand

(e.g., fuse boxes), which are included with  $D_i = 0$ . An example network layout as retrieved from the geographic information system is illustrated in Figure III-1.

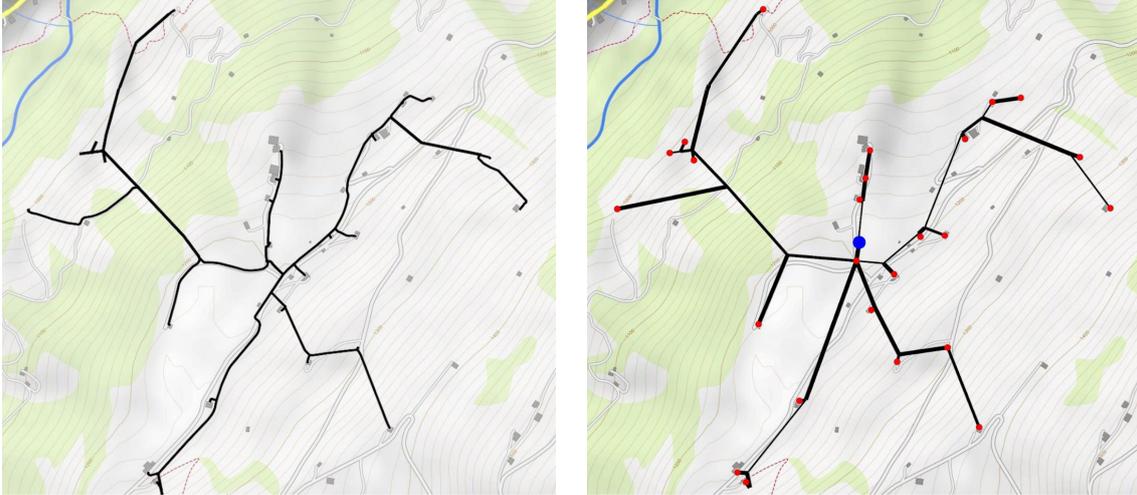


Figure III-1: Data extraction from the geographic information system: (left) real-world network (with buildings, roads and landscape), and (right) extracted layout and capacities (small dots: loads, large dot: transformer, capacity indicated by line thickness).

## 4.2 Parameters and Unit Conversions

Below, all parameters and units are shown that have been used for the real-world case study. All parameters for the experiment are set to conventional values from practice in agreement with practitioners from the partnering company. For the edge capacities, the values correspond to commercially available copper cables, identical to the ones used in the original networks. The capacity of these cables is given as cross section in  $\text{mm}^2$ . Capacities are chosen from the set  $A = \{50 \text{ mm}^2, 70 \text{ mm}^2, 95 \text{ mm}^2, 120 \text{ mm}^2, 150 \text{ mm}^2, 185 \text{ mm}^2, 240 \text{ mm}^2, 400 \text{ mm}^2, 800 \text{ mm}^2\}$ . The industry standard of  $D^{\text{peak}} = 21 \text{ kW}$  is used for the peak demand per load. Based on the peak load, the demand  $D_i$  is determined using the coincidence factor  $\gamma$  (Dickert and Schegner, 2010). Here the formula  $\gamma(|I_j|) = \gamma_{\text{lim}} + (1 - \gamma_{\text{lim}})|I_j|^{-1/2}$  is used, with  $\gamma_{\text{lim}} = 0.1$ , which provides a conservative estimate. All flows  $F_{ij}$  are given in the unit of kW.

Now, the appropriate unit conversions are added to the objective function and the constraints. For the objective function, the cost constants follow the original cost composition and are based on discussions with network design experts from the part-

nering company.  $c_c = 34.62 \frac{\text{CHF}}{\text{m}}$  and  $c_m = 0.1882 \frac{\text{CHF}}{\text{m mm}^2}$  is used. For the objective function, this yields

$$\sum_{(i,j)^k \in E} x_{ij}^k [34.62 \frac{\text{CHF}}{\text{m}} l_{ij} + 0.1882 \frac{\text{CHF}}{\text{m mm}^2} l_{ij} a_{ij}^k]. \quad (\text{III-50})$$

The line sizing constraint for the real-world case study reads

$$1.08 \frac{\text{kW}}{\text{mm}^2} \sum_{k \in \{1, \dots, |A|\}} x_{ij}^k a_{ij}^k \geq F_{ij}, \quad \forall i, j \in \{0, \dots, n-1\}, \quad (\text{III-51})$$

The conversion factor  $1.08 \frac{\text{kW}}{\text{mm}^2}$  has been extracted by a linear fit from the data sheets of the electrical cables used. All voltage drops along the grid lines must stay below 3% of the network voltage of 0.4 kV. This is a conservative value in line with industry norms (cf. CENELEC, 2010). For a three-phase 0.4 kV electricity distribution network, the constraint for the voltage drops reads

$$\sum_k x_{ij}^k \frac{a_{ij}^k}{l_{ij}} (U_i - U_j) = \frac{\sqrt{3}}{0.4 \text{ kV}} 0.0181 \frac{\Omega \text{mm}^2}{\text{m}} F_{ij}, \quad \forall i, j \in \{0, \dots, N-1\}, \quad (\text{III-52a})$$

$$U_i \geq 0.4 \text{ kV} - 3\% 0.4 \text{ kV}, \quad \forall i \in \{0, \dots, N-1\}, \quad (\text{III-52b})$$

$$U_0 = 0.4 \text{ kV}. \quad (\text{III-52c})$$

All distances are given in m. Reactive power of the loads is assumed to be zero. The value  $\rho = 0.0181 \frac{\Omega \text{mm}^2}{\text{m}}$  specifies the resistivity of the grid lines and has been extracted from the data sheet of the electrical cables used.

Furthermore, all parameters used as part of the heuristics are identical to the previous computational experiments, i. e., the random improvement heuristic uses  $s_{\max} = 3N$ ; the increased and decreased branching heuristics use the parameter  $d_{\max} = 4$ ; the VNS algorithm uses  $d_{\max} = 3$ ,  $s_{\text{VNS}} = 5$ , and  $s_{\max} = N$  for the local search; the Tabu Search algorithm uses  $s_{\text{tabu}} = 10N$ ; for instances with  $N \leq 20$ , the length of the tabu list is set to 5; for all other instances, it is set to 10.

## 4.3 Results

### 4.3.1 Solution Quality

In two out of the 74 cases, some heuristics were not able to find a feasible solution. That is, the heuristics for generating the network layout provide a layout for which the constraint for the voltage drops cannot be fulfilled even when all capacities are set to the maximum possible value. Note that such infeasible solutions can evidently be prevented when using any combination of the starred network algorithm. This is ensured by the theoretical properties of the heuristic (cf. Remark 3). The starred network algorithm produces no infeasible network layouts, while the MST creates three infeasible network layouts.

Table III-8 compares the cost of the solutions generated by the heuristics to the real-world costs from conventional network design. All costs are given in Swiss Francs (CHF). The cost have been averaged over those instances where a feasible solution was found by every heuristic. The proposed solution methods are able to solve the real-world instances effectively for all instance sizes. Overall, the two meta-heuristics (VNS and Tabu Search) perform best, with Tabu Search yielding the lowest cost for every group (i. e., for every instance size). Among the local improvement heuristics, three combinations result in a similar cost (MST combined with the INB heuristic, MST combined with the randomized network reconfiguration, and greedy network construction combined with the INB heuristic). Out of these three combinations, each one is cheaper than the others at least once.

For smaller instances with fewer than 30 loads, the best heuristic returns networks that reduce the cost from conventional network design by 39 % to 41 %. This corresponds to absolute savings of more than CHF 61,000 per network for these networks. For larger instances, the observed saving increases, both in relative and absolute terms. For example, for networks with  $N \in [30, 49]$ , the benefit compared to the real-world networks is 63 % to 69 %. For instances with 50 or more loads, the cost

Initial Layout	Layout Improvement	$N$	[10, 19]	[20, 29]	[30, 39]	[40, 49]	[50, 59]	[60, 69]
MST	No heuristic		61,811	112,437	73,068	98,776	96,224	133,422
	Randomized Reconfiguration		60,089	105,422	70,531	89,195	85,408	125,081
	INB		59,538	108,178	71,626	91,030	85,046	123,605
	DNB		60,017	110,118	72,007	94,779	88,933	125,173
	VNS		59,409	104,444	70,232	89,391	84,044	124,137
	Tabu Search		59,392	103,923	68,922	87,153	82,839	122,489
Starred Network Algorithm	No heuristic		223,199	403,038	437,828	519,045	541,489	788,838
	Randomized Reconfiguration		144,761	274,906	257,302	307,324	327,064	514,532
	DNB		62,130	109,115	75,917	90,106	86,814	126,804
	VNS		113,721	223,879	195,636	231,363	259,405	415,220
	Tabu Search		95,334	172,342	141,496	187,267	194,277	305,096
Greedy Network Construction	No heuristic		61,109	112,625	75,760	91,849	90,006	127,400
	Randomized Reconfiguration		60,305	105,669	72,021	88,582	84,943	122,744
	INB		59,923	108,025	72,404	87,634	84,457	120,688
	DNB		59,701	109,355	74,504	89,729	85,172	121,105
	VNS		59,715	104,842	71,067	87,642	84,910	121,368
	Tabu Search		59,329	104,009	70,672	87,592	82,872	120,771
Real-world network		103,562	181,483	223,028	236,488	300,642	386,180	
<i>Number of networks in sample group</i>			14	19	9	8	14	7

Table III-8: Comparison of network cost for networks with various number of loads  $N$ . Figures shown are only averages over those instances whereby a feasible network could be found by every combination. After a maximum calculation time of 24 hours per network, the cheapest network is returned.

benefit increases even further, with absolute cost savings of up to CHF 0.26 million per network.

### 4.3.2 Runtime

The runtimes for the real-world experiments are shown in Table III-9. Overall, similar observations than in the computational experiments can be made. Even for the largest network instances, runtimes are still below 24 hours. This demonstrates that the proposed solution approaches are computationally tractable.

### 4.3.3 Summary

In summary, the real-world experiments prove the applicability of the heuristics to real-world problem instances and point toward significant cost savings. In the experiments, the proposed solution methods yield relative cost savings of over 39%. The cost savings for larger networks can be as high as CHF 0.26 million. This shows how

Initial layout	Layout improvement	<i>N</i>	[10, 19]	[20, 29]	[30, 39]	[40, 49]	[50, 59]	[60, 69]
MST	No heuristic		0.14	0.57	1.95	6.73	9.06	17.21
	Randomized Reconfiguration		45.54	270.90	1512.41	4736.11	9366.02	23341.30
	INB		200.43	1420.85	3292.12	15612.27	47439.29	80704.57
	DNB		64.94	421.93	1020.67	3202.57	10024.94	22746.30
	VNS		298.55	1434.05	7998.43	19409.59	40760.33	60429.04
	Tabu Search		155.61	900.46	4502.16	16400.05	29648.26	69967.41
Starred Network Algorithm	No heuristic		0.00	0.01	0.01	0.01	0.01	0.04
	Randomized Reconfiguration		1.92	7.86	45.64	63.96	124.35	262.58
	DNB		147.90	837.79	2752.44	9343.13	23503.21	57484.61
	VNS		10.88	38.86	168.69	457.48	553.46	1031.22
	Tabu Search		33.96	120.58	705.98	1640.82	2787.01	5760.48
Greedy Network Construction	No heuristic		0.07	0.36	1.06	2.62	4.17	8.68
	Randomized Reconfiguration		26.44	187.53	832.33	2858.80	8512.50	14791.68
	INB		342.30	4038.22	11742.90	43463.82	71366.42	86413.25
	DNB		98.60	561.90	1930.40	7553.07	19357.20	44611.04
	VNS		198.39	898.71	2907.58	13096.60	35008.78	59516.61
	Tabu Search		109.79	708.08	2793.07	8521.77	26686.28	45355.99
<i>Number of networks in sample group</i>			14	19	9	8	14	7

Table III-9: Comparison of the average runtime per network using different capacity optimization methods for real-world experiment. Maximum calculation time has been set to 24 hours per network.

the proposed solution methods to the CAVLP can provide effective decision support to network planners at utility companies.

## 5 Summary

In this chapter, a decision problem for optimizing electricity distribution networks has been introduced. This problem uses voltage drops and load coincidence, while, at the same time, adhering to other physical constraints (e. g., radiality). This yields the NP-hard capacitated arborescence with voltage drops and load coincidence problem. Based on the theoretical properties of the problem, heuristics for solving the CAVLP have been derived. This includes theoretical bounds for some. The effectiveness of the heuristics is demonstrated based on computational experiments and real-world electricity networks.

The presented work has several implications for both management and research. For management, the proposed solution approaches are able to provide more cost-effective network layouts with cost savings over the actual network layouts of more than 39%. This corresponds to cost savings of up to CHF 0.26 million per network. Furthermore, it is shown that cost savings increase with network sizes. Evidently,

conventional network design is especially suboptimal for larger networks. This points towards large opportunities for decision support.

In the next chapter, the effect of various functions of the coincidence factor on the network cost and layout is investigated. The decision problem from this chapter is used to study the effect of (A) direct adaptations to climate change, (B) an increasing share of EVs, and (C) a further decentralized energy generation. Please note, that because the CAVLP can also be adapted to solve scenarios with excess feed-in by reverting the flows, it can be used to evaluate the monetary impact of increased PV diffusion in (C).

## 6 Additional Information

### 6.1 Proof of NP-Hardness for Alternative Problem Formulation

There are two ways of proving NP-hardness using the alternative problem formulation. The first one uses the line sizing constraint in Equation (III-2). Proposition 1 is proven by reduction. It is shown that the CAVLP is a generalized form of the NP-hard capacitated shortest spanning arborescence rooted at  $r$  (CSSA $_r$ ) problem (Papadimitriou, 1978; Toth and Vigo, 1995). More precisely, it is shown that the CSSA $_r$  problem is a special case of the CAVLP with  $Q = \infty$  and  $a_{ij}^k = a$  and a uniform coincidence factor  $\gamma(|S|) \equiv 1$ .

By setting  $Q = \infty$ , we can ignore the voltage drops in Equations (III-4a) to (III-4c). By setting  $a_{ij}^k = a$ , the line sizing constraint in Equation (III-2) reduces to  $F_{ij} \leq a$  for all  $i, j \in \{0, \dots, n-1\}$ . The flow  $F_{ij}$  corresponds to the sum of all demands downstream to an edge, i. e., demands of all vertices in  $\Gamma_j$ . Because of the radial layout, the flow  $F_{ij}$  decreases for edges  $(i, j)^k$  that are located further downstream. Thus, if  $F_{ij} \leq a$  holds true for an edge connected to the source, it is automatically fulfilled for all subsequent edges. Therefore, the line sizing constraint only needs to be

checked for edges connected to the source, i. e., for edges  $(i, j)^k$  with the depth  $d(j) = 1$ . The line sizing constraint simplifies to  $F_{0j}(\{x_{ij}^k\}) \leq a$  for all  $j \in \{0, \dots, n-1\}$ . With the definitions of  $F_{ij}$  and the subgraph  $\Gamma_j$ , this becomes

$$\sum_{i \in \Gamma_j} D_i \leq a, \quad \forall j \text{ with } d(j) = 1. \quad (\text{III-53})$$

Equation (III-53) states that the sum of demands  $D_i$  in all subgraphs  $\Gamma_j$  directly connected to the source must not exceed  $a$ . Equation (III-53) is identical to the constraint of the  $\text{CSSA}_r$  problem which aims to find the shortest spanning arborescence under this condition (cf. Toth and Vigo, 1995).

As this reduction is clearly of polynomial time, this proves that the CAVLP is NP-hard.  $\square$

The second way of proving NP-hardness involves the constraint for the voltage drop. We prove this even for the special case of a uniform coincidence factor  $\gamma(|S|) \equiv 1$ , which reduces the complexity of the problem. For this proof, we use the new notation for the neighborhood  $\mathcal{N}_j(\{x_{ij}^k\}, d)$ , which is the set of all vertices that can be reached from a vertex  $j$  within  $d$  hops (in direction of the flow). The definition of the flow  $F_{ij}$  becomes

$$F_{ij} \stackrel{\text{def}}{=} \sum_{i \in \mathcal{N}_j(\{x_{ij}^k\}, N_{\max})} D_i, \quad (\text{III-54})$$

where we define  $N_{\max} = N - 1$  as the maximum number of hops it can possibly take to reach any vertex downstream of another vertex.

An approach similar to that one found in the proof of Proposition 1 is used. It is shown that the CAVLP is a generalized form of the  $\text{CSSA}_r$  problem (see Toth and Vigo, 1995). First of all, we bring the constraint for the voltage drops in a different

form. According to Equation (III-4a), the voltage drop along any edge  $(i, j)^k$  is given by

$$(U_i - U_j) = \sum_k x_{ij}^k l_{ij} \frac{F_{ij}}{a_{ij}^k}. \quad (\text{III-55})$$

Because the voltage drops are accumulating along any path  $p \in P(\Gamma)$  from source to any leaf, the constraint for the voltage drop can be rewritten as

$$\sum_k x_{ij}^k l_{ij} \frac{F_{ij}}{a_{ij}^k} \leq Q, \quad \forall p \in P(\Gamma). \quad (\text{III-56})$$

Again, we prove NP-hardness even when there is only one unitary line type, i. e.,  $a_{ij}^k = a$ . Then, the only difference between the  $\text{CASSA}_r$  problem and the CAVLP lies in the constraint for the voltage drops, i. e., Equation (III-56). In the  $\text{CSSA}_r$  problem, the demand of all vertices in each sub-tree leaving the source must not exceed a given maximum capacity  $Q$ . In our notation, this reads

$$\sum_{i,j \text{ s. t. } (i,j)^k \in p} D_j \leq Q, \quad \forall p \in P(\Gamma). \quad (\text{III-57})$$

As we want to show a reduction of the  $\text{CSSA}_r$  problem to the CAVLP, we rewrite the voltage drop from Equation (III-56) by using the definition for  $F_{ij}$  in Equation (III-54), yielding

$$\sum_{i,j \text{ s. t. } (i,j)^k \in p} \sum_{k \in \mathcal{N}_j(N_{\max})} l_{ij} D_k \leq Q. \quad (\text{III-58})$$

From Equation (III-58), we obtain the constraint of the  $\text{CSSA}_r$  problem by making two simplifications. First, we set  $N_{\max} = 0$ , which means that, for calculating  $F_{ij}$ , we ignore all subsequent vertices and  $F_{ij} = D_j$ . In other words, the second sum

in Equation (III-58) reduces to  $\sum_{k \in \mathcal{N}_j(0)} D_k l_{ij} = l_{ij} D_j$ . Second, we assume unified distances by setting  $l_{ij} = 1$  and, as a result, Equation (III-58) becomes

$$\sum_{i,j \text{ s.t. } (i,j)^k \in p} D_j \leq Q. \quad (\text{III-59})$$

This is identical to the constraint of the  $\text{CSSA}_r$  problem. Consequently, the  $\text{CSSA}_r$  problem is a special case of the CAVLP with unified distances  $l_{ij} = 1$  and  $N_{\max} = 0$ .

This shows that the CAVLP is a generalized form of the  $\text{CSSA}_r$  problem, which is known to be NP-hard. All reductions are of polynomial time, which concludes this proof.  $\square$

## 6.2 Sensitivity Analysis of Capacity Optimization Methods

In this section, a sensitivity analysis of the capacity optimization methods for cost and runtime is conducted. These results confirm that the PECA heuristic yields consistent results, independent from the network layout and demand. We remember that the improvement heuristics trigger a capacity optimization for each candidate layout and, owing to this, the capacity optimization is responsible for a considerable part of the runtime of an improvement heuristic.

The PECA heuristic is tested against an exact solution using the Gurobi Optimizer 7.5.2 and two greedy heuristics. For this testing, three different methods to generate network layouts are used: (1) the MST, (2) the greedy network construction, and (3) a method, whereby all edges are generated completely at random (random layout generation). The latter method generates a random Prüfer sequence of length  $N - 2$ , from which the network layout is created. For all these methods, the capacity optimization methods are tested in a low demand case and a high demand case. In total, this leads to six different test settings. In sum, these support the choice of the PECA heuristic, since, independent of how the network layout is constructed, it finds capacities that are close to the optimal solution yet in considerably less time.

Below, the two greedy heuristics are first presented in more detail. Second and third, the results for the low and high demand experiments are presented.

### 6.2.1 Greedy Capacity Reinforcement

The greedy capacity reinforcement heuristic optimizes the capacities by steadily increasing them until the constraint for the voltage drops is fulfilled. It starts with minimal edge capacities, successively identifies the edge  $(i, j)$  with the highest voltage drop (this corresponds to the weak spot of the network), and then increases the capacity of this edge. This heuristic resembles common industry practices in electricity network expansion. For instance, it is used by the Swiss electricity power company that provided us with real-world data.

The heuristic works in five steps. In step 1, all capacities  $a_{ij}$  are initialized to the minimum values that fulfill the line sizing constraint. At this point, it is not guaranteed that the constraint for the voltage drops in Equation (III-40) is fulfilled. In step 2, the heuristic calculates the voltage drops  $\Delta_{U_{ij}}$  for each edge of the network. In step 3, the heuristic determines the set of paths  $P' \subseteq P$  where the constraint for the voltage drops is violated. If all paths fulfill the constraints, the heuristic terminates and returns  $\{a_{ij}\}$ . In step 4, the heuristic considers all paths  $p \in P'$  and reinforces the edge  $(i, j) \in p$  with the highest voltage drop  $\Delta_{U_{ij}}$  by increasing its capacity  $a_{ij}$  to the next larger capacity, i. e., from  $a_{ij}^k$  to  $a_{ij}^{k+1}$ . In step 5, the voltage drop  $\Delta_{U_{ij}}$  for this edge is recalculated, since increasing the capacity reduces the voltage drop. With these updated capacities, the heuristic returns to step 3. The runtime of this algorithm depends on the network layout. For example, in case of a starred network, the runtime is  $O(N)$ . For other layouts, runtimes are higher because both the number of paths and the depth of the paths increase with the network size  $N$  (cf. Steele et al., 1987).

### 6.2.2 Greedy Capacity Reduction

The greedy capacity reduction heuristic is the counterpart to the greedy capacity reinforcement heuristic and proceeds in the opposite direction. It identifies the edge

$(i, j)$  with the lowest voltage drop and reduces its capacity in order to save material cost. It works in five steps. In step 1, all capacities  $a_{ij}$  are initialized to the maximum value possible and the heuristic creates a list  $L$  containing all edges. In step 2, the heuristic calculates the voltage drops  $\Delta_{U_{ij}}$  for each edge. In step 3, the heuristic identifies the edge  $(i, j) \in L$  with the lowest voltage drop. If  $L$  is empty, the heuristic terminates and returns  $\{a_{ij}\}$ . In step 4, the heuristic decreases  $a_{ij}$  by one decrement from the value  $a_{ij}^k$  to  $a_{ij}^{k-1}$ . In step 5, the heuristic evaluates whether this reduction violates the constraints related to line sizing and voltage drops. If they are violated, the capacity of the edge  $(i, j)$  is reset to  $a_{ij}^k$  and the edge is removed from  $L$ . If the constraints are still fulfilled,  $\Delta_{U_{ij}}$  for this edge is recalculated and the heuristic returns to step 3.

The runtime of this heuristic scales similarly to the greedy capacity reinforcement heuristic. In practical applications, however, it entails a disadvantage with regard to runtime: close to the leaves, networks typically consist of many edges with low capacity. Thus, its runtime can be expected to be slower than the runtime of the greedy capacity reinforcement heuristic as more iterations are required.

### 6.2.3 Sensitivity Analysis – Low Demand (Solution Quality)

Below, the average cost per network and the coefficient of variation is shown for various instance sizes  $N$  below. The PECA heuristic is largely on par with the exact solver (i. e., Gurobi MIP solver), even for large networks. As expected, the PECA heuristics outperforms the other heuristics—greedy capacity reinforcement and greedy capacity reduction—due to its theoretical properties.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		7.43 (0.23)	12.51 (0.16)	16.46 (0.14)	19.55 (0.11)	22.25 (0.12)	24.59 (0.11)	27.77 (0.13)	30.16 (0.11)	31.62 (0.12)	33.72 (0.11)
Greedy capacity reduction		7.43 (0.23)	13.04 (0.19)	17.65 (0.17)	20.89 (0.13)	23.66 (0.13)	25.86 (0.12)	28.81 (0.13)	31.07 (0.10)	32.46 (0.10)	34.30 (0.09)
PECA		7.41 (0.23)	12.39 (0.15)	16.12 (0.13)	18.94 (0.10)	21.44 (0.11)	23.61 (0.10)	26.45 (0.12)	28.60 (0.11)	30.04 (0.11)	31.90 (0.10)
Exact solver (Gurobi MIP)		7.40 (0.22)	12.37 (0.15)	16.05 (0.13)	18.85 (0.10)	21.34 (0.11)	23.48 (0.10)	26.30 (0.12)	28.41 (0.11)	29.84 (0.11)	31.63 (0.10)

Table III-10: Comparison of network cost using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using the MST algorithm on randomly generated locations for each  $N$ . Figures shown are averages over 100 networks. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16GB of RAM.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy Capacity Reinforcement		7.53 (0.24)	12.57 (0.16)	16.47 (0.13)	19.28 (0.10)	21.99 (0.09)	24.55 (0.09)	26.60 (0.09)	29.01 (0.07)	30.74 (0.07)	32.79 (0.07)
Greedy Capacity Reduction		7.53 (0.24)	12.77 (0.17)	16.95 (0.15)	19.89 (0.11)	22.74 (0.09)	25.37 (0.10)	27.35 (0.09)	29.87 (0.08)	31.67 (0.08)	33.61 (0.07)
PECA		7.51 (0.24)	12.48 (0.16)	16.21 (0.12)	18.83 (0.09)	21.41 (0.08)	23.70 (0.09)	25.60 (0.08)	27.83 (0.07)	29.50 (0.06)	31.31 (0.06)
Gurobi MIP solver (exact solution)		7.51 (0.24)	12.46 (0.15)	16.17 (0.12)	18.75 (0.09)	21.32 (0.08)	23.58 (0.08)	25.46 (0.07)	27.65 (0.06)	29.31 (0.06)	31.08 (0.06)

Table III-11: Comparison of network cost using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using the greedy network construction. Figures shown are averages over 100 networks.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		12.00 (0.29)	28.04 (0.21)	44.92 (0.20)	64.29 (0.16)	84.80 (0.15)	103.83 (0.15)	122.39 (0.16)	146.31 (0.15)	170.41 (0.16)	184.69 (0.15)
Greedy capacity reduction		12.05 (0.30)	28.40 (0.21)	45.48 (0.20)	63.68 (0.15)	83.08 (0.14)	100.10 (0.13)	117.28 (0.14)	137.67 (0.12)	156.74 (0.11)	172.49 (0.11)
PECA		11.99 (0.29)	27.60 (0.21)	43.44 (0.20)	61.56 (0.15)	79.93 (0.13)	96.97 (0.13)	113.72 (0.14)	133.51 (0.12)	151.31 (0.11)	166.24 (0.11)
Exact solver (Gurobi MIP)		11.97 (0.29)	27.51 (0.21)	43.20 (0.19)	61.13 (0.14)	79.00 (0.13)	95.62 (0.13)	111.84 (0.13)	131.06 (0.12)	148.93 (0.11)	163.62 (0.11)

Table III-12: Comparison of network cost using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using a randomly generated layout with randomly generated locations for each  $N$ . Figures shown are averages over 100 networks. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16GB of RAM.

### 6.2.4 Sensitivity Analysis – Low Demand (Runtime)

In terms of runtime, the PECA heuristic outperforms the exact solver considerably. Furthermore, the PECA heuristic has a slightly slower runtime than the greedy capacity reinforcement heuristic (but better cost performance as shown above). The greedy capacity reinforcement heuristic is computationally more efficient than the greedy capacity reduction heuristic. This matches the earlier expectations as solutions are likely to entail many edges with low capacity edges close to the leaves, which are more easily identified by the greedy capacity reinforcement heuristic compared to the greedy capacity reduction heuristic.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.02	0.03	0.04
Greedy capacity reduction		0.00	0.01	0.02	0.05	0.08	0.12	0.18	0.28	0.37	0.47
PECA		0.00	0.00	0.00	0.01	0.02	0.04	0.07	0.13	0.18	0.22
Exact solver (Gurobi MIP)		0.01	0.02	0.03	0.06	0.09	0.12	0.16	0.23	0.27	0.33

Table III-13: Comparison of the average runtime using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using the MST algorithm on randomly generated locations for each  $N$ . Figures shown are averages over 100 networks. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16GB of RAM.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
<b>Greedy Capacity Reinforcement</b>		0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.02
<b>Greedy Capacity Reduction</b>		0.00	0.01	0.02	0.05	0.08	0.13	0.18	0.26	0.34	0.46
<b>PECA</b>		0.00	0.00	0.00	0.01	0.01	0.02	0.03	0.05	0.06	0.11
<b>Gurobi MIP solver (exact solution)</b>		0.01	0.02	0.03	0.05	0.08	0.12	0.15	0.20	0.25	0.32

Table III-14: Comparison of average runtimes using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using the greedy network construction. Figures shown are averages over 100 networks. Most runtimes are fairly small (below 1 second), hence, the coefficient of variation is omitted. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16 GB of RAM.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.02	0.04	0.05
Greedy capacity reduction		0.00	0.01	0.02	0.05	0.08	0.12	0.18	0.24	0.33	0.48
PECA		0.00	0.00	0.00	0.01	0.02	0.03	0.05	0.07	0.11	0.17
Exact solver (Gurobi MIP)		0.01	0.02	0.03	0.06	0.09	0.12	0.17	0.21	0.26	0.37

Table III-15: Comparison of the average runtime using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using a randomly generated layout with randomly generated locations for each  $N$ . Figures shown are averages over 100 networks. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16GB of RAM.

### 6.2.5 Sensitivity Analysis – High Demand (Solution Quality)

Below average cost for the high demand case is shown. For each  $N$ , average over 100 networks is reported. If infeasible network layouts were generated, these layouts were discarded because it is impossible for any of the capacity optimization methods to find capacities for such layouts.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		8.24 (0.24)	13.77 (0.18)	18.61 (0.15)	22.52 (0.15)	25.83 (0.13)	29.45 (0.12)	31.73 (0.12)	34.25 (0.11)	37.04 (0.11)	38.90 (0.12)
Greedy capacity reduction		8.52 (0.27)	14.65 (0.19)	19.54 (0.15)	23.01 (0.13)	26.02 (0.11)	28.94 (0.09)	30.93 (0.09)	33.28 (0.08)	35.53 (0.08)	37.13 (0.08)
PECA		8.14 (0.24)	13.25 (0.17)	17.61 (0.15)	21.13 (0.15)	23.97 (0.12)	26.87 (0.10)	28.80 (0.10)	30.93 (0.08)	32.96 (0.08)	34.40 (0.09)
Exact solver (Gurobi MIP)		8.10 (0.24)	13.18 (0.17)	17.51 (0.15)	20.99 (0.15)	23.78 (0.12)	26.67 (0.10)	28.53 (0.10)	30.61 (0.09)	32.63 (0.08)	34.15 (0.09)

Table III-16: Comparison of network cost using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using the MST algorithm on randomly generated locations for each  $N$ . Figures shown are averages over 100 networks. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16GB of RAM.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		8.29 (0.25)	14.17 (0.17)	18.18 (0.13)	22.16 (0.13)	25.22 (0.12)	28.83 (0.12)	31.31 (0.10)	34.00 (0.10)	37.11 (0.11)	39.78 (0.10)
Greedy capacity reduction		8.45 (0.26)	14.85 (0.18)	18.85 (0.14)	22.48 (0.12)	25.65 (0.11)	28.82 (0.11)	31.34 (0.09)	33.59 (0.08)	36.42 (0.08)	38.70 (0.08)
PECA		8.19 (0.25)	13.67 (0.17)	17.25 (0.12)	20.70 (0.11)	23.50 (0.10)	26.54 (0.11)	28.84 (0.09)	30.99 (0.08)	33.59 (0.09)	35.78 (0.08)
Exact solver (Gurobi MIP)		8.16 (0.25)	13.62 (0.17)	17.13 (0.12)	20.54 (0.11)	23.31 (0.10)	26.30 (0.11)	28.55 (0.09)	30.65 (0.08)	33.27 (0.09)	35.34 (0.08)

Table III-17: Comparison of network cost using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using the greedy network construction. Figures shown are averages over 100 networks.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		13.63 (0.33)	32.14 (0.22)	53.20 (0.20)	75.18 (0.19)	95.91 (0.17)	120.03 (0.15)	143.08 (0.14)	168.74 (0.15)	190.08 (0.12)	212.02 (0.13)
Greedy capacity reduction		13.92 (0.35)	32.41 (0.22)	51.54 (0.18)	71.12 (0.16)	89.15 (0.14)	109.21 (0.12)	128.65 (0.11)	147.85 (0.12)	168.24 (0.10)	185.06 (0.11)
PECA		13.42 (0.33)	30.80 (0.21)	49.39 (0.18)	68.12 (0.16)	85.36 (0.15)	105.08 (0.13)	123.79 (0.12)	141.65 (0.12)	162.13 (0.10)	178.30 (0.11)
Exact solver (Gurobi MIP)		13.40 (0.33)	30.60 (0.21)	49.00 (0.18)	67.32 (0.16)	84.28 (0.15)	103.62 (0.13)	121.94 (0.12)	140.05 (0.12)	159.30 (0.10)	175.37 (0.11)

Table III-18: Comparison of network cost using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using a randomly generated layout with randomly generated locations for each  $N$ . Figures shown are averages over 100 networks. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16GB of RAM.

### 6.2.6 Sensitivity Analysis – High Demand (Runtime)

Below the corresponding runtimes for the high demand case are shown. For each  $N$ , the average over 100 networks is reported.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.05	0.06
Greedy capacity reduction		0.00	0.01	0.02	0.04	0.07	0.09	0.15	0.20	0.26	0.34
PECA		0.00	0.00	0.01	0.02	0.04	0.05	0.08	0.10	0.15	0.19
Exact solver (Gurobi MIP)		0.01	0.02	0.04	0.06	0.10	0.13	0.17	0.22	0.27	0.33

Table III-19: Comparison of the average runtime using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using the MST algorithm on randomly generated locations for each  $N$ . Figures shown are averages over 100 networks. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16GB of RAM.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		0.00	0.00	0.00	0.00	0.01	0.01	0.02	0.03	0.03	0.04
Greedy capacity reduction		0.00	0.01	0.02	0.04	0.07	0.10	0.15	0.26	0.28	0.36
PECA		0.00	0.00	0.00	0.01	0.02	0.04	0.06	0.11	0.12	0.18
Exact solver (Gurobi MIP)		0.01	0.02	0.03	0.06	0.09	0.13	0.17	0.28	0.28	0.34

Table III-20: Comparison of average runtimes using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using the greedy network construction. Figures shown are averages over 100 networks. Most runtimes are fairly small (below 1 second), hence, the coefficient of variation is omitted. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16 GB of RAM.

Capacity Optimization	$N$	5	10	15	20	25	30	35	40	45	50
Greedy capacity reinforcement		0.00	0.00	0.00	0.01	0.01	0.02	0.03	0.04	0.05	0.06
Greedy capacity reduction		0.00	0.01	0.02	0.04	0.07	0.10	0.14	0.19	0.25	0.31
PECA		0.00	0.00	0.01	0.01	0.02	0.04	0.06	0.08	0.10	0.13
Exact solver (Gurobi MIP)		0.01	0.02	0.04	0.06	0.09	0.12	0.17	0.20	0.26	0.31

Table III-21: Comparison of the average runtime using different capacity optimization methods for networks with various number of vertices  $N$ . Network layouts have been generated using a randomly generated layout with randomly generated locations for each  $N$ . Figures shown are averages over 100 networks. The calculations were performed on an Intel Core i7-7600 CPU at 2.8 GHz and 16GB of RAM.

## CHAPTER IV

# QUANTIFYING THE EFFECT OF LOAD COINCIDENCE ON NETWORK DESIGN

In the previous chapter, the monetary benefit of including load coincidence in distribution network design have been demonstrated. Load coincidence has been included by using a standard function for the coincidence factor from Dickert and Schegner (2010). Now, the objective is to better understand how changes in this function affect network cost and layouts.

In Chapter I, three trends have been presented that can result in a more coinciding energy usage, namely (A) direct adaptations to climate change, (B) an increasing share of EVs, and (C) a further decentralized energy generation. In the present chapter, these trends are translated into scenarios by assigning different functions for the coincidence factor to each of the scenarios. The Tabu Search metaheuristic is used to generate networks for each of these scenarios. Both the model and the solution approach are slightly modified to give a more generalizable problem and to allow for the trouble-free creation of simulated network instances with  $N = 100$ , even for high demand situations. The set of 74 real-world network instances is used again as a case study. The results indicate that changes in the coincidence factor can have significant effect on network costs and layouts. Depending on the scenario and network size, network cost can more than double.

In summary, this chapter has the following contributions:

1. A more generalized version of the CAVLP is presented. It is proven that this new problem is also NP-hard.
2. The solution methods are refined to be able to solve the generalized problem and to allow for trouble-free creation of large network instances.
3. For the first time, it is investigated how differences in the coincidence factor affect network cost and layouts. A better understanding of the interplay between coincidence factors and network cost and layouts provides a practical value-add as it guides network planners in solving real-world problems in a more cost-effective manner. It also provides valuable insights to other stakeholders (e. g., policy makers).
4. The results of the experiments show that particularly the simultaneous charging of EVs can cause a significant increase in network cost (up to 84 % in the computational experiments and up to 159 % in the case study). However, the effects from simultaneously running AC units or PV feed-in cannot be neglected either (up to 27 % cost increase in the computational experiments and up to 108 % cost increase in the case study).

## 1 The Model

A generalized version of the CAVLP is used in this chapter. We call this problem the generalized capacitated arborescence with voltage drops and load coincidence problem (gCAVLP). The notation is provided in Table IV-1. This notation is almost identical to the the notation of the CAVLP.

Symbol	Description	Unit/range
$G$	Directed multigraph	$G = (V, E)$
$V$	Set of all vertices	
$N$	Number of vertices	$N =  V $
$i, j$	Indices of vertices	$i, j = 0, \dots, N - 1$
$D_i$	Demand of vertex $i$	$D_0 = 0, D_{i \neq 0} = D > 0$ .
$E$	Set of all directed edges	
$k$	Index of line type	$k = 1, \dots,  A $
$(i, j)^k$	Directed edge from $i$ to $j$ ; $k$ denoting its type	$(i, j)^k \in E$
$A$	Set of edge capacities depending on line type $k$ (in ascending order)	
$a_{ij}^k$	Edge capacity	$a_{ij}^k > 0, a_{ij}^k \in A$
$C$	Set of current carrying capacities depending on line type $k$ (in ascending order)	
$c_{ij}^k$	Current carrying capacity of an edge	$c_{ij}^k > 0, c_{ij}^k \in C$
$N_j$	Set of all vertices that can be reached from $j$	
$\Gamma$	Graph representing one solution of the problem	$\Gamma = (V, E')$ , with $E' = \{(i, j)^k \in E \mid x_{ij}^k = 1\}$
$\Gamma_j$	Subgraph of $\Gamma$ including $j$ and all edges and vertices reachable from $j$	
$ \Gamma_j $	Number of vertices in $\Gamma_j$	
$\gamma( \Gamma_j )$	Coincidence factor (discount factor depending on number of vertices)	$0 < \gamma( \Gamma_j ) \leq 1$
$\bar{D}_i$	(Un-discounted) sum of all demands in $\Gamma_j$	
$d(i)$	Depth of vertex $i$ , i. e., number of hops to reach $i$ from the source in $\Gamma$	
$F_{ij}^k$	Flow through edge $(i, j)^k$	$F_{ij}^k > 0$
$l_{ij}^k$	Length of edge $(i, j)^k$	$l_{ij}^k \in \mathbb{R}^+$
$P$	Set of all paths from source vertex 0 to any leaf vertex	Set of edge sequences
$p$	Specific path from source vertex 0 to a leaf vertex	Edge sequence, $p \in P$
$c_c$	Construction costs	Monetary unit per distance
$c_m$	Material costs	Monetary unit per distance per capacity unit
$U_i$	Voltage at vertex $i$	$U_i > 0$
$U$	Voltage at transformer	$U > 0, U_0 = U$
$U_{crit}$	Critical voltage level	$U_{crit} > 0$
$Q$	Threshold value for voltage drop	$Q = U - U_{crit}$
$x_{ij}^k$	Decision variable for edge from vertex $i$ to $j$ with capacity $a_{ij}^k$	$x_{ij}^k \in \{0, 1\}$

Table IV-1: Notation for the gCAVLP.

In the original CAVLP, one capacity  $a_{ij}^k$  was introduced. This capacity corresponded to the cross section of the grid line. Three assumptions were implied here, namely that (a) material cost is proportional to  $a_{ij}^k$ , (b) the maximum allowed electric current is proportional to  $a_{ij}^k$ , and that (c) voltage drops are inversely proportional to  $a_{ij}^k$ . For the gCAVLP, assumptions (a) and (c) are kept. Assumption (a) is supported by the cost data used for the case studies. Assumption (c) is supported by the physical nature of voltage drops. Assumption (b) is an approximation of reality that arises when the relationship between cross section and maximum allowed current is linearized. In reality, however, the maximum allowed current is limited by the heat dissipation of the electrical cable used. Heat dissipation is proportional to the cable's

outer surface area and thus is proportional to  $\sqrt{a_{ik}^k}$ . For this reason, a new capacity  $c_{ij}^k$  is introduced. This property is often referred to as current carrying capacity, representing the maximum allowed flow for each line type  $k$ . The gCAVLP is then given by

$$\min \sum_{(i,j)^k \in E} x_{ij}^k [l_{ij} c_c + l_{ij} c_m a_{ij}^k] \quad (\text{IV-1})$$

$$\text{s. t.} \quad \sum_{k \in \{1, \dots, |A|\}} x_{ij}^k c_{ij}^k \geq F_{ij}, \quad \forall i, j \in V, \quad (\text{IV-2})$$

$$\begin{aligned} \sum_j F_{ji} - \sum_j F_{ij} &= \gamma (|\Gamma_i|) D_i - \\ &\sum_j \sum_k x_{ij}^k ([\gamma (|\Gamma_j|) - \gamma (|\Gamma_i|)] \bar{D}_j), \quad \forall i \in V \setminus \{0\}, \end{aligned} \quad (\text{IV-3a})$$

$$|\Gamma_i| = 1 + \sum_j \sum_k x_{ij}^k |\Gamma_j|, \quad \forall i \in V, \quad (\text{IV-3b})$$

$$\bar{D}_i = D_i + \sum_j \sum_k x_{ij}^k \bar{D}_j, \quad \forall i \in V, \quad (\text{IV-3c})$$

$$|\Gamma_0| = N - 1, \quad (\text{IV-3d})$$

$$\bar{D}_0 = \sum_j D_j, \quad (\text{IV-3e})$$

$$\sum_i \sum_k x_{ij}^k = 1, \quad \forall j \in V \setminus \{0\}, \quad (\text{IV-3f})$$

$$\sum_k x_{ij}^k \frac{a_{ij}^k}{l_{ij}} (U_i - U_j) = F_{ij}, \quad \forall i, j \in V, \quad (\text{IV-4a})$$

$$U_i \geq U_{\text{crit}}, \quad \forall i \in V, \quad (\text{IV-4b})$$

$$U_0 = U. \quad (\text{IV-4c})$$

This problem formulation is identical to the problem in Equations (III-1) to (III-4c), with the exception of Equation (IV-2). In this equation, the current carrying capacity  $c_{ij}^k$  is used instead of  $a_{ij}^k$ .

## 1.1 Solution Properties

Because the gCAVLP is a generalized version of the original CAVLP, the gCAVLP is NP-hard. This is stated by the following proposition.

**Proposition 6** *The gCAVLP is NP-hard.*

*Proof.* This is proven by reduction. By setting  $c_{ij}^k = a_{ij}^k$  for all  $i, j, k$ , the gCAVLP reduces to the original CAVLP. This problem is NP-hard (see Proposition 1).  $\square$

In a similar way, it can be shown that the solution properties derived in Section 1.6 and Section 2.3 are still valid for the gCAVLP.

**Proposition 7** *The solution properties in Proposition 3, 4 and 5, Remarks 2 and 3, as well as Corollaries 1 and 2 hold true for the gCAVLP.*

*Proof.* The proof starts by proving this for Remarks 2 and 3. These remarks regard the MST and the starred network. Regarding the MST, we follow the proof of Remark 2: In low demand situations, the constraints for both line sizing in Equation (IV-2) and voltage drops in Equation (IV-4b) are automatically fulfilled and can be ignored. Therefore, Remark 2 also holds true for the gCAVLP because the two problems are identical. Regarding the starred network, we need to make the reasonable assumption that the higher the cross section, the higher the current carrying capacity. This is always the case in reality, as grid lines with larger cross sections can also carry larger currents. Therefore, by choosing the maximum value for  $a_{ij}^k$ , we automatically chose the maximum value for  $c_{ij}^k$ . We can then follow the proof for Remark 3.

Proposition 3 assumes that the constraint for the voltage drop is binding. Therefore, we can ignore the current carrying capacity  $c_{ij}^k$  in the generalized problem. If this is the case, the gCAVLP is identical to the original CAVLP and Proposition 3 holds true. Propositions 4 and 5 directly result from Proposition 3 and are therefore automatically fulfilled.

Corollaries 1 and 2 directly follow from Propositions 3 and 4 and are thus also fulfilled, which concludes this proof  $\square$

## 2 Optimization Methods

### 2.1 General Optimization Approach

Similar to before, the gCAVLP is split into the two sub-problems, namely (A) generating the network layout and (B) capacity optimization. (A) generating the network layout determines which connections  $\{x_{ij}\}$  to choose. (B) capacity optimization then determines the capacities  $\{a_{ij}\}$  (and thus also  $\{c_{ij}\}$ ) for this particular layout. The computational experiments in Section 3 and the real-world case study in Section 4 have shown that the optimization method with the best cost performance is the combination MST + Tabu Search. However, this combination has one shortcoming: As demands get higher and networks get larger, the MST produces too many infeasible networks (see Section 3.3).

As a remedy, the following approach is developed. The optimization approach uses the MST as a starting network layout whenever possible (i. e., if the MST is a feasible layout), and reverts to other layouts with an increased branching in the rest of cases. To generate these networks with increased branching, the Esau-Williams (EW) algorithm (Esau and Williams, 1966) is utilized. This algorithm is described in detail in Section 2.2.2. In short, it is a greedy algorithm that provides near-optimal solutions to the capacitated minimum spanning tree (CMST) problem (Bruno and Laporte, 2002). The CMST problem is identical to the CSSA<sub>r</sub> problem but with undirected edges (Voß, 2001). It aims to find the cycle-free network connecting all  $N$  vertices with the shortest total edge length. Thereby, each of the subtrees directly connected the source must contain at most  $K$  vertices. As shown below, for  $K = 1$ , the Esau-Williams algorithm always returns the starred network (Remark 5). For  $K = N - 1$ , the solution is close to the MST.

In the proposed optimization approach, the EW algorithm is used if the MST algorithm produces an infeasible layout. Then, the EW algorithm is used to find the network layout for  $K = \frac{N}{2}$ . If this is still infeasible, the EW algorithm is used with  $K = \frac{N}{4}$ ,  $K = \frac{N}{8}$ , etc., until a feasible layout is found. Essentially, this allows us to tune the branching of the starting layout depending on the demand situation. This approach to determine the initial layout is exemplified in Figure IV-1, where the MST and various layouts are displayed. These other layouts have been found using the EW algorithm.

After the initial layout is found, this layout is used as input for the Tabu Search algorithm to find a solution for the overall problem. This general optimization approach is detailed in Algorithm 8.

---

**Algorithm 8** General Optimization Approach

---

```

1: Create MST by using MINIMUMSPANNINGTREEALGORITHM
2:  $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$ 
3:  $n = 1$ 
4: while network is infeasible do
5:   Perform ESAUWILLIAMSALGORITHM( $K \leftarrow \lceil \frac{N}{2^n} \rceil$ )  $\triangleright \frac{N}{2^n}$  is rounded up, such that  $K \geq 1$  and integer
6:    $\{a_{ij}\} \leftarrow \text{CAPACITYOPTIMIZATIONMETHOD}(\{x_{ij}\})$ 
7:    $n \leftarrow n + 1$ 
8: Perform TABUSEARCH( $\{x_{ij}\}$ )
9: return  $X^*$ 

```

---

The optimization approach as described in Algorithm 8 produces infeasible solutions only if there is no solution at all. This is described by the following remark.

**Remark 4** *If feasible solutions to the gCAVLP exist, the solution approach in Algorithm 8 will return a feasible solution.*

*Proof.* In the worst case, i. e., if no feasible solution is found for several iterations, the optimization approach eventually sets  $K \leftarrow 1$  in line 5. As shown later (Remark 5), the EW algorithm returns the starred network layout, if  $K$  is set to 1. Therefore, the starting layout is the starred network. For the original CAVLP, we already saw that if feasible solutions exist, the starred network is one of them (Remark 3). Because



this remark also holds true for the gCAVLP (Proposition 7), we know that the solution approach will return a feasible solution, if it exists.  $\square$

## 2.2 Generating an Initial Layout

Below, the two algorithms used to create an initial network layout are described. These are the minimum spanning tree algorithm and the Esau-Williams algorithm.

### 2.2.1 Minimum Spanning Tree Algorithm

Prim's algorithm (Prim, 1957) is used to determine the MST. The algorithm and its properties are described in Section 2.1.1.

### 2.2.2 Esau-Williams Algorithm

The Esau-Williams algorithm is a greedy algorithm for solving the CMST problem (Jothi and Raghavachari, 2004). This problem is identical to the  $CSSA_r$  problem, with the exception of having undirected edges (while the edges of the  $CSSA_r$  problem are directed (Voß, 2001)). Therefore, the directions of all edges must be fixed in the last step of the optimization process below. The objective of the CMST problem is to find an MST with the additional constraint that each of the subtrees directly connected the source vertex must contain at most  $K$  vertices.

The general principle of the EW algorithm is to subsequently merge subtrees. This is done until no benefit can be reached by merging two subtrees without violating the capacity constraint. The capacity constraint for merging two subtrees  $T_i$  and  $T_j$  is given by  $|T_j| + |T_i| \leq K$ . The algorithm uses two lists, which are initialized in line 2. The first list  $b_i$  tracks the benefit of reconnecting the subtree  $T_i$ . The second list  $n_i$  tracks the best option for  $T_i$  to be connected to. For each vertex  $i$ ,  $n_i$  corresponds to the nearest neighbor to  $i$  not contained in  $T_i$  not violating the capacity constraint. The algorithm is presented in Algorithm 9.

**Algorithm 9** Esau-Williams Algorithm

---

```

1: Initialize  $x_{ij} \leftarrow 0$  for all  $i, j$ 
2: Initialize  $b_i, n_i \leftarrow 0$  for all  $i$ 
3:  $x_{0j} \leftarrow 1$  for all  $j$  and  $x_{ij} \leftarrow 0$  otherwise ▷ Create starred layout
4: while 1 do
5:    $b_{\max} \leftarrow 0$  ▷ Set maximum benefit  $b_{\max}$  to 0
6:   for each vertex  $i \in V$  directly connected to the source do
7:     Find nearest neighboring vertex  $j \notin \Gamma_i$ , with  $|\Gamma_j| + |\Gamma_i| \leq K$ 
8:      $n_i \leftarrow j$  ▷ Store nearest neighbor for later
9:      $b_i \leftarrow l_{0i} - l_{ij}$  ▷ Calculate benefit value for connecting  $i$  and  $j$ 
10:     $b_{\max} \leftarrow \max_i(b_i)$  ▷ Determine maximum benefit
11:    if  $b_{\max} \leq 0$  then break ▷ Break if no further improvement can be made
12:     $m \leftarrow i$ , such that  $b_i = b_{\max}$ 
13:     $n \leftarrow n_m$ 
14:     $x_{mn} \leftarrow 1, x_{0m} \leftarrow 0$  ▷ Connect  $m$  to  $n$  and delete old edge
15: for each leaf vertex  $j$  do
16:   for each edge  $(m, n)$  in path from 0 to  $j$  do
17:      $x_{mn} \leftarrow 1, x_{nm} \leftarrow 0$  ▷ Check direction of all edges
18: return  $\{x_{ij}\}$ 

```

---

The EW algorithm starts by creating a starred network layout in line 3. The main part of the algorithm (lines 4 to 14) is performed until no further improvement can be made. The possibility of improvement is tracked by the maximum benefit  $b_{\max}$ , which is set to 0 in line 5. In lines 6 to 8, the algorithm loops through each vertex  $i$  directly connected to the source and finds the nearest neighbor  $n_i$ . By doing so, the algorithm takes into account the capacity constraint in line 7. The potential benefit for connecting  $i$  and  $n_i$  (i. e., the reduction in length) is calculated in line 9. The maximum possible benefit for reconnecting any vertex is determined in line 10. If no positive benefit can be made (line 11), the algorithm jumps directly to line 15. If this is not the case, it makes the reconnection resulting the highest benefit in lines 12 to 14. The EW algorithm is designed to solve the CMST problem with undirected edges. Thus, at the end of the algorithm, the directions of all edges need to be checked. It needs to be ensured that they point away from the source. This is done in lines 15 to 17. The algorithm returns the (directed) network layout  $\{x_{ij}\}$  in line 18.

The runtime of the EW algorithm is in  $O(N^2 \log N)$  and is known to be among the most efficient algorithms to solve the CMST problem (Jothi and Raghavachari, 2004).

The following remark concerns an important property of the EW algorithm, namely that it returns the starred network layout for  $K = 1$ . This is crucial for proving Remark 4.

**Remark 5** *For  $K = 1$ , the Esau-Williams algorithm always returns the starred network layout.*

*Proof.* The EW algorithm starts with a starred network layout in line 3. If  $K$  is set to 1, no nearest neighbor fulfilling the capacity constraint can be found in line 7. This is because  $F_i \geq 1$ , for all  $i$ . Therefore, no two subgraphs  $|F_i|$  and  $|F_j|$  can be merged without violating the capacity constraint  $|F_j| + |F_i| \leq 1$ . As a consequence,  $b_{\max}$  has the value 0 and the algorithm breaks in line 11.  $\square$

## 2.3 Improving an Existing Network Layout

The computational experiments in Section 3 and the case study in Section 4 show that the Tabu Search metaheuristic outperforms other solution methods. This is independent of the network size and the demand situation (low demand vs. high demand). Therefore, the Tabu Search algorithm presented in Section 2.2.5 is used to improve the network layout.

## 2.4 Optimizing the Capacities

The gCAVLP is split into the two sub-problems: (A) generating the network layout and (B) capacity optimization. Below, the problem formulation for (B) capacity optimization is given. It determines the capacities  $\{a_{ij}\}$  (and thus also  $\{c_{ij}\}$ ) for a given layout. The main difference between sub-problem (B) and the full gCAVLP (i. e., the combination of the two sub-problems) is the fact that capacity optimization only considers a subset of edges  $E' \subset E$ , namely these edges where  $x_{ij} = 1$  has been determined in the first sub-problem. As a consequence, the flows are now given and  $F_{ij}$  is no longer

an auxiliary decision variable. Capacity optimization is formulated as a binary integer problem:

$$\min \sum_{(i,j)^k \in E'} x_{ij}^k l_{ij} a_{ij}^k \quad (IV-5)$$

$$\text{subject to } \sum_{k \in \{1, \dots, |A|\}} x_{ij}^k c_{ij}^k \geq F_{ij}, \quad \forall i, j \in \{0, \dots, N-1\}, \quad (IV-6)$$

$$\sum_k x_{ij}^k = 1, \quad \forall (i, j)^k \in E', \quad (IV-7)$$

$$\sum_{(i,j)^k \in p} x_{ij}^k l_{ij} \frac{F_{ij}}{a_{ij}^k} \leq U - U_{\text{crit}} = Q, \quad \forall p \in P. \quad (IV-8)$$

The problem above is identical to the one in Equations (III-38) to (III-40), with one exception: In Equation (IV-6), the capacity  $a_{ij}^k$  is substituted with the current carrying capacity  $c_{ij}^k$ .

To solve the capacity optimization problem, the PECA algorithm from Section 2.3.2 is used. This algorithm has shown to be very efficient (see Section 6.2). The algorithm needs to be adjusted slightly because of the introduction of the current carrying capacity  $c_{ij}^k$  in the gCAVLP. In line 3 in Algorithm 7, the capacities  $a_{ij}$  need to be set to

$$a_{ij} \leftarrow \min_k \left\{ a_{ij}^k \mid c_{ij}^k \geq F_{ij} \text{ and } a_{ij}^k \geq \frac{1}{Q} \sum_{(i,j) \in p} l_{ij} F_{ij} \right\}.$$

Similar changes need to be applied in line 16. Here the line sizing constraint changes from  $a_{ij}^{k-1} \geq F_{ij}$  to

$$c_{ij}^{k-1} \geq F_{ij}.$$

The remaining algorithm stays untouched.

### 3 Scenarios for Coincidence Factors

#### 3.1 Background and Assumptions

The objective of this chapter is to investigate the effect of various coincidence factors on network design. The functions for the coincidence factors are motivated by new technologies that shift demand patterns (e. g., PV systems and EVs). Four different scenarios are developed, which are detailed below, namely a base scenario (Section 3.2.1), a scenario with increased coincidence (Section 3.2.2), a worst case scenario for simultaneous charging of EVs (Section 3.2.3), and a worst case scenario for PV feed-in (Section 3.2.4).

Because it is the objective to investigate changes in load coincidence isolated from changes in peak loads themselves, in all scenarios, the individual peak loads are kept at an identical level. This choice is made on purpose to demonstrate how large the influence of load coincidence on its own can be.

#### 3.2 Building the Scenarios

The formulas used for the coincidence factor  $\gamma(|I_j|)$  in all scenarios are based on Dickert and Schegner (2010) and originates from Rusck (1956). For better readability, in this section,  $|I_j|$  is replaced by  $N$ .  $\gamma(N)$  is thus the discount factor that can be applied to a group of  $N$  loads. The formula for  $\gamma(N)$  reads

$$\gamma(N) = \gamma_{\text{lim}} + (1 - \gamma_{\text{lim}}) N^{-1/2}. \quad (\text{IV-9})$$

Below, this formula is quickly motivated. The derivation follows the one from Chapter IV (A) in Dickert and Schegner (2010).

We start with the assumption that the individual loads during the peak demand are normal distributed. With this assumption, we can calculate the standard deviation  $\sigma_i$  of an individual load curve  $D_i(t)$ , which yields

$$\sigma_i = \sqrt{\int (D_i(t) - \bar{D})^2 dt}. \quad (\text{IV-10})$$

Summing up multiple normally distributed loads results in a new load curve, which, again, is normally distributed. The standard deviation of the resulting load curve is

$$\sigma = \sqrt{\sum_i \sigma_i^2}. \quad (\text{IV-11})$$

If the loads are similar, we can follow the argumentation of Rusck (1956), namely that the difference between the maximum value of each load and its mean value is proportional to the standard deviation. This yields

$$D_{\max}(N) - \sum_i \bar{D}_i = \sqrt{\sum_i (D_{\max,i} - \bar{D}_i)^2}, \quad (\text{IV-12})$$

with  $D_{\max}(N)$  being the coincident peak demand of  $N$  loads.

If we assume  $N$  loads that are all normally distributed with mean load  $\bar{D}$  and peak load  $D$ , this can be rewritten to give

$$D_{\max}(N) = N \cdot \bar{D} + (D - \bar{D}) \cdot \sqrt{N}. \quad (\text{IV-13})$$

Dividing Equation (IV-13) by  $N$  results in Equation (IV-9), with  $\gamma_{\text{lim}} = D_{\max}(N)/N$ .

Finally, it should be noted that a variety of different formulas for the coincidence factor exist. For a specific region, they are often determined in empirical studies (e. g., Nickel and Braunstein, 1981). In this thesis, the choice Equation (IV-9) is made because it is a very general formula and because there are existing parametrizations for various customer types that can be utilized to develop the scenarios.

### 3.2.1 Base Scenario

For the base scenario, the identical formula formula for the coincidence factor as in Section 3 and Section 4 is used. The parameter  $\gamma_{\text{lim}}$  is set to 0.1, which yields

$$\gamma(N) = 0.1 + (1 - 0.1) N^{-1/2}. \quad (\text{IV-14})$$

According to Dickert and Schegner (2010), this parametrization describes customer types, where all domestic appliances are electric. Customers might have an electric stove and/or water boilers. However, customers should not have flow-type water heaters or electric space heating (and/or cooling). This scenario corresponds to a situation that is typical for countries like Germany or Switzerland, which is why it has been chosen for the real-world case study in Section 4.

### 3.2.2 Increased Coincidence

For the scenario of increased coincidence,  $\gamma_{\text{lim}} = 0.7$  is chosen. For the coincidence factor, this yields

$$\gamma(N) = 0.7 + (1 - 0.7) N^{-1/2}. \quad (\text{IV-15})$$

The value of  $\gamma_{\text{lim}} = 0.7$  corresponds to customers that exhibit electric space heating (Dickert and Schegner, 2010). It is therefore used as a proxy for situations, in which large loads are being used by multiple households in a relatively coinciding manner. Examples other than space heating might be space cooling (using AC units) or charging of EVs. This scenario thus corresponds to geographic regions where electric space heating or cooling is typical, or a scenario of increased EV penetration. For an EV penetration rate of close to 100 %, a separate worst case scenario is developed below.

### 3.2.3 Worst Case Scenario: Simultaneous Charging of EVs

The uncontrolled simultaneous charging of EVs poses enormous challenges to distribution networks (Richardson et al., 2012; Wieland et al., 2015). Gaul et al. (2017) analyzed 450,000 charging sessions at public charging stations in 2014. They found out that in 300 hours of the year, all connected cars were charging at full power. The authors thus conclude that a coincidence factor of 1.0 must be assumed when describing the underlying distribution network. For the worst case scenario of simultaneous charging of EVs, it is thus assumed that

$$\gamma(N) \equiv 1.0. \quad (\text{IV-16})$$

Transferring this to a residential setting, one could argue that this corresponds to a hypothetical scenario, where all households in a neighborhood are in possession of at least one EV, and, where the charging occurs sometimes simultaneously (e.g., overnight). At first, this seems like an extreme scenario. However, one could also argue that the scenario underestimates the stress put on the network, because it assumes that the peak demands of the individual households remain unchanged. In practice, EV charging is added on top of the existing power demand of a household. If, for instance, the residents arrive from work in the evening and charge their EV simultaneously to cooking dinner, this would certainly increase the peak demand of the household. Note that in this thesis the objective is to investigate the influence of changes in the coincidence factor independently from changes in peak demand. Therefore, the individual peak demands are identical in all scenarios for better compatibility.

### 3.2.4 Worst Case Scenario: PV Feed-In

Developing the appropriate formula for integrating the uncontrolled feed-in of PV energy into the distribution network works slightly different than the integration of charging EVs. The reason is that (at least in residential settings) the peak power of

a PV system of a single household does not exceed the peak demand of this household. Therefore, naively viewed, adding a PV system should not add additional strain on the network. However, if multiple households feed-into the network, these feed-ins are very likely to coincide (because of the underlying weather conditions). This coinciding nature of energy feed-in contrasts the stochastic nature of energy demand. When building the coincidence factor for this scenario, these two factors need to be combined.

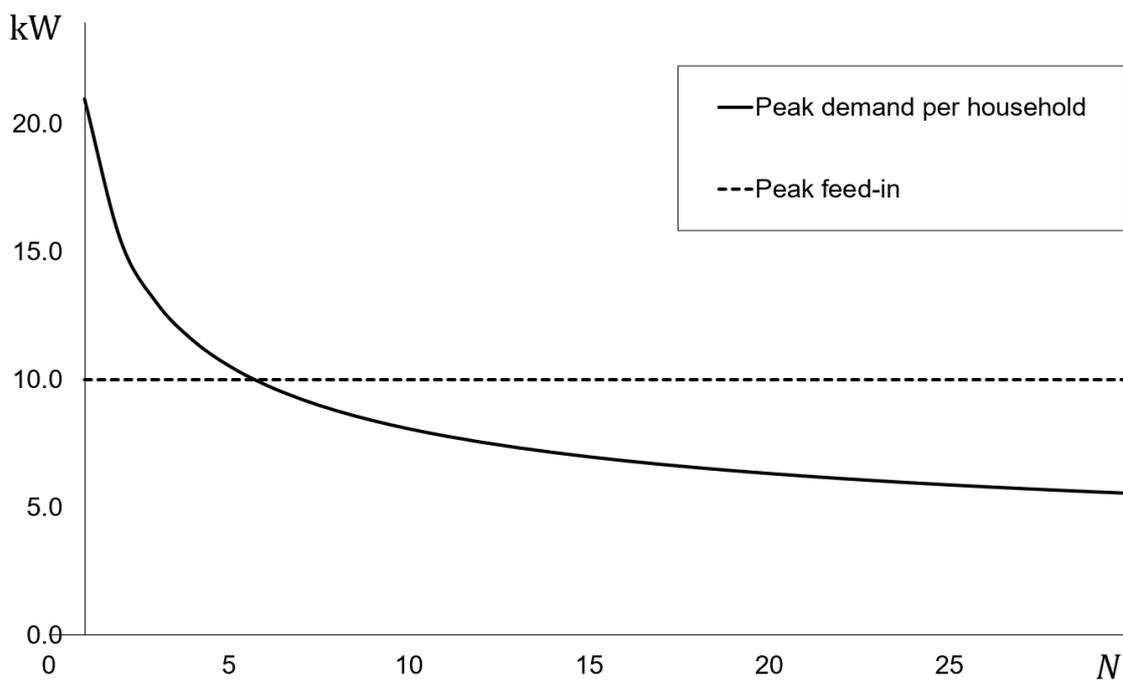


Figure IV-2: Comparison of peak demand per household (solid line) and peak feed-in per household (dashed line) for households with  $D_{\text{peak}} = 21 \text{ kW}$  and  $D_{\text{peak}}^{\text{PV}} = 10 \text{ kW}$ . At  $N \approx 5$ , the curves cross. Therefore, for  $N \leq 5$ , the peak demand is dominant, for  $N > 5$ , the peak feed-in is dominant.

To illustrate this, consider several households with a peak demand of  $D_{\text{peak}} = 21 \text{ kW}$ , each with a PV system with a peak power of  $D_{\text{peak}}^{\text{PV}} = 10 \text{ kW}$ . The curves for coinciding peak demand per household  $N$  and coinciding feed-in power per household  $N$  are shown in Figure IV-2. The former follows the base scenario from Section 3.2.1, the latter assumes  $\gamma(N) \equiv 1.0$ , but at a lower power of 10 kW. If the two curves are combined, we need to take into account the maximum of both curves. The

coinciding demand is thus dominant for small  $N$ , while the PV feed-in is dominant for large  $N$ . For the scenario, the formula for the coincidence factor becomes

$$\gamma(N) = \max \left\{ 0.1 + (1 - 0.1)N^{-1/2}, \frac{D_{\text{peak}}^{\text{PV}}}{D_{\text{peak}}} \right\}. \quad (\text{IV-17})$$

### 3.2.5 Summary of Scenarios

The four scenarios are depicted in Figure IV-3. The coincidence factor  $\gamma(N)$  is shown as a function of  $N$ . Clearly, the base case exhibits the smallest value for  $\gamma$ , for all  $N$ . The worst case scenario for simultaneous charging of EVs has the highest value for  $\gamma$ , for all  $N$ . The scenario for increased coincidence converges towards a coincidence factor of  $\gamma_{\text{lim}} = 0.7$ . Therefore, its value is above the worst case scenario for PV feed-in, which remains constant at  $\gamma(N) = 0.476$ , for  $N > 5$ .

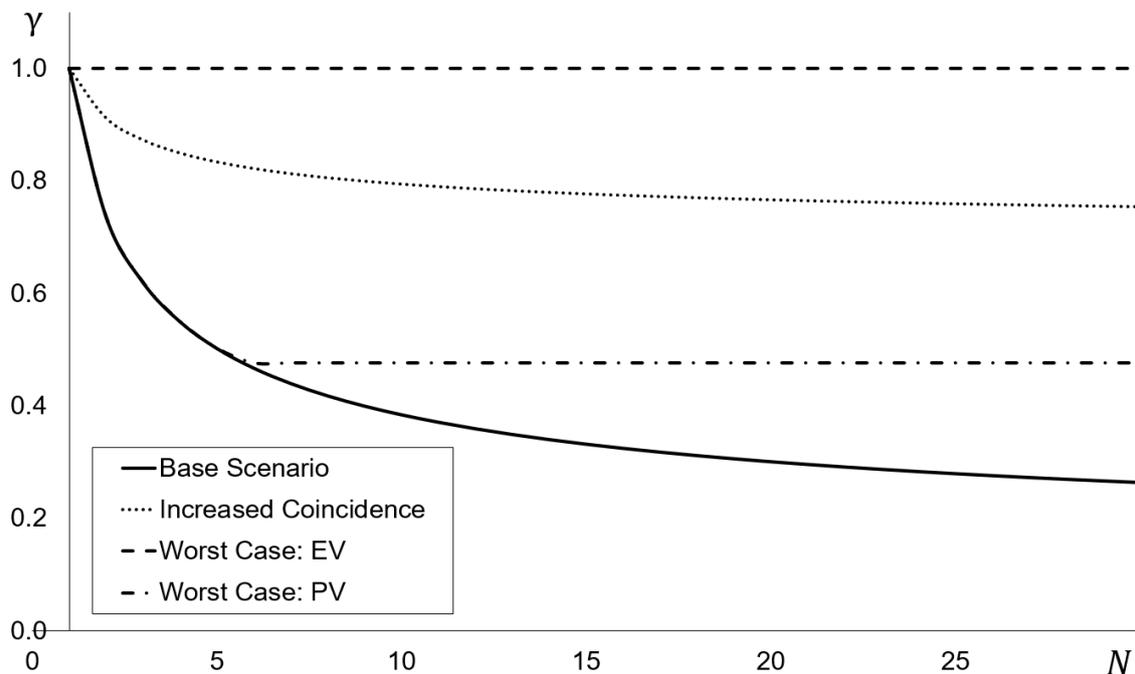


Figure IV-3: Comparison of the coincidence factors  $\gamma(N)$  assumed for various scenarios: Base scenario (solid line), increased coincidence (dotted line), worst case scenario: simultaneous charging of EVs (dashed line), worst case scenario: PV feed-in (dash-dotted line).

## 4 Computational Experiments

The effect of changes in the coincidence factor is evaluated using randomly generated problem instances of various sizes. The experimental setup is presented in Section 4.1. The results are reported in Section 4.2. Three cases are considered in the experiments: The main case (medium density) exhibits a household density (i. e., loads per ha) that is comparable to the density found in the real-world networks of the case study. Further, a case with low household density and a case with high household density are added.

### 4.1 Experimental Setup

The parameters for the computational experiments is presented in Table IV-2. As opposed to the previous computational experiments in Section 3, the identical units, cost functions, capacities, and parameters as in the real-world case study are used. This is for the following reason. Previously, the computational experiments served to prove the effectiveness of the proposed solution methods. Now, only one solution method is used and the objective is to approximate the (real-world) effect of load coincidence.

Network size	Network area	Voltage drop threshold	Peak demands
$N \in \{20, 40, \dots, 100\}$	10.0 households per ha (high density) 1.0 households per ha (medium density) 0.1 households per ha (low density)	$U_{\text{crit}} = 0.97 U$	$D^{\text{peak}} = 21 \text{ kW}$

Table IV-2: Parameters for computational experiments.

The experiments are conducted on networks ranging from  $N = 20$  to  $N = 100$ . This is made possible by the improved solution methods (see Section 2). The locations are randomly placed on a square grid. Depending on the case, the area of the square is chosen, such that 10 households (high density), 1 household (medium density), or 0.1 households (low density) are placed on it. The density in the main case (medium density) corresponds to the density found in the real-world sample.

The threshold for the voltage drop is set to  $U_{\text{crit}} = 0.97 U$ . The industry standard of  $D^{\text{peak}} = 21 \text{ kW}$  is used for the peak demand per load as before. For the edge capacities, values corresponding to commercially available copper cables are used. These are identical to the ones used in the original networks. The capacity of these cables is given as cross section in  $\text{mm}^2$ . Capacities  $a_{ij}^k \in A$  are chosen from the set  $A = \{50 \text{ mm}^2, 70 \text{ mm}^2, 95 \text{ mm}^2, 120 \text{ mm}^2, 150 \text{ mm}^2, 185 \text{ mm}^2, 240 \text{ mm}^2, 400 \text{ mm}^2, 800 \text{ mm}^2\}$ . The current carrying capacity per line type is given in A. The current carrying capacities  $c_{ij}^k \in C$  can be chosen from  $C = \{185 \text{ A}, 228 \text{ A}, 274 \text{ A}, 313 \text{ A}, 352 \text{ A}, 398 \text{ A}, 464 \text{ A}, 510 \text{ A}, 671 \text{ A}\}$ .

The unit conversions of the objective function and the constraints are similar than in Section 4.2. The objective function reads

$$\sum_{(i,j)^k \in E} x_{ij}^k \left[ 34.62 \frac{\text{CHF}}{\text{m}} l_{ij} + 0.1882 \frac{\text{CHF}}{\text{m mm}^2} l_{ij} a_{ij}^k \right]. \quad (\text{IV-18})$$

The line sizing constraint reads

$$0.4 \text{ kV} \sum_{k \in \{1, \dots, |A|\}} x_{ij}^k c_{ij}^k \geq F_{ij}, \quad \forall i, j \in \{0, \dots, n-1\}. \quad (\text{IV-19})$$

Note that the flows are given in kW, while  $c_{ij}^k$  is given in A. The constraint for the voltage drops reads

$$\sum_k x_{ij}^k \frac{a_{ij}^k}{l_{ij}} (U_i - U_j) = \frac{\sqrt{3}}{0.4 \text{ kV}} 0.0181 \frac{\Omega \text{mm}^2}{\text{m}} F_{ij}, \quad \forall i, j \in \{0, \dots, N-1\}, \quad (\text{IV-20a})$$

$$U_i \geq 0.4 \text{ kV} - 3\% 0.4 \text{ kV}, \quad \forall i \in \{0, \dots, N-1\}, \quad (\text{IV-20b})$$

$$U_0 = 0.4 \text{ kV}. \quad (\text{IV-20c})$$

The Tabu Search algorithm uses  $s_{\text{tabu}} = 10 N$ . The length of the tabu list has been set to 10 for networks of size  $N \leq 60$ . For the larger networks ( $N > 60$ ), a tabu list of length 20 is chosen.

## 4.2 Results

### 4.2.1 Main Case: Medium Household Density

Table IV-3 displays the results for the main case, i. e., the case with medium household density of 1.0 households per ha. The table shows the network cost in CHF on the left and the corresponding network lengths in km on the right. Additionally, the number of branches is shown in brackets on the right hand side.

Scenario	Cost in CHF					Length in km (no. of branches)				
	<i>N</i>	20	40	60	80	100	20	40	60	80
Base Scenario	62,693	154,960	262,903	388,040	552,708	1.419 (2.5)	3.058 (2.9)	4.764 (3.5)	6.640 (4.8)	8.901 (6.0)
Increased Coincidence	67,464	178,680	314,139	487,631	704,483	1.450 (2.6)	3.217 (3.4)	5.226 (5.1)	7.459 (6.9)	10.225 (8.7)
Worst Case: Charging of EVs	75,400	220,424	409,903	667,388	1,015,165	1.535 (3.3)	3.716 (5.5)	6.405 (8.6)	9.799 (11.9)	14.434 (15.5)
Worst Case: PV Feed-In	68,290	178,133	314,797	486,914	705,379	1.459 (2.7)	3.254 (3.7)	5.214 (5.3)	7.501 (6.9)	10.225 (8.6)

Table IV-3: Left: Comparison of network cost in CHF for various number of loads  $N$ . Reported is the cost averaged across 50 instances. The cells are shaded based on average cost. Right: Comparison of network length in km with number of branches in parenthesis. The cells are shaded based on average length.

Regarding network cost, it can be observed that, as expected, for all scenarios, the network cost grows as the networks get larger. Further, the base scenario exhibits the lowest cost for all network sizes, while the worst case scenario for simultaneous charging of EVs has the highest cost. The other two scenarios (increased coincidence and worst case: PV feed-in) are fairly comparable in cost. This means that changes in the function for the coincidence factor do not have a significant cost effect, as long as those changes are in between  $\gamma_{lim} \approx 0.5$  and  $\gamma_{lim} \approx 0.7$  (which is approximately the magnitude of change between the two scenarios at hand). The cost difference between these two scenarios in Table IV-3 can be ascribed to the stochastic nature of the Tabu Search algorithm. Further, the relative cost gap between the base scenario and the other three scenarios increases with  $N$ : For  $N = 20$ , the gap is between 8 and 20 %, for  $N = 100$ , the gap is between 27 and 84 %.

Regarding network length and branching, both figures increase with  $N$ . Similar to before, the gap between the base scenario and the other three scenarios also increases the larger the networks get. For the network length, this gap increases from 2–8 % for the smallest networks ( $N = 20$ ) to 15–65 % for the largest networks

( $N = 100$ ). The branching grows even more: The smallest networks ( $N = 20$ ) are 4–28 % more branched-out compared to the base scenario, while the largest networks ( $N = 100$ ) exhibit branching that is 43–159 % higher than the base scenario. This means that, for  $N = 100$ , the networks in the worst case scenario for simultaneous charging of EVs have more than 2.5-times as many branches as networks in the base scenario.

For the sake of completeness, the runtimes are addressed below. Table IV-4 shows the average runtimes for the main case. First, note that in general, runtimes increase as the networks get larger. For the smallest instance ( $N = 20$ ), runtimes for all scenarios stay below one minute per network instance. For the largest instances ( $N = 100$ ), runtimes are less than one hour per instance. These runtimes are more than sufficient for real-world applications. Second, the runtimes for the base scenario are longest, followed by the worst case scenario for PV feed-in and the scenario with increased coincidence. The worst case scenario for simultaneous charging of EVs exhibits the smallest runtimes. This is in line with observations made in Sections 3.2 and 3.3, namely that the smaller the branching, the higher the runtimes. This can be explained by the nature of the Tabu Search heuristic. In each iteration, this heuristic forms a cycle within the network. It then deletes the most cost-efficient edge within that cycle by looping through all edges of the cycle. As the branching gets smaller, the branches themselves become longer. This translates into longer cycles and therefore longer runtimes.

Scenario	Time in seconds					
	$N$	20	40	60	80	100
Base Scenario		31.19	421.02	1,442.44	3,170.52	5,688.35
Increased Coincidence		29.91	295.63	964.38	2,392.83	4,034.09
Worst Case: Charging of EVs		19.51	181.66	656.40	1,283.75	2,441.95
Worst Case: PV Feed-In		30.70	310.91	990.22	2,387.38	4,286.23

Table IV-4: Comparison of runtimes in seconds for various number of loads  $N$ . Reported is the averaged runtime across 50 instances.

In summary, the experiments show that changes in load coincidence have a significant effect on network cost, length and branching. Network cost can be up to 85 %

higher compared to the base scenario, which is accompanied by a 62 % increase in network length and a 2.5-fold higher branching. Simultaneous charging of EVs has the most severe effect on network cost. The cost effect of the scenario of increased coincidence is comparable to the effect that is expected from an uncontrolled PV feed-in.

### 4.2.2 Low and High Household Density

Table IV-5 displays both the results for the case with a low household density of 0.1 households per ha (top) and the results for the case with a high household density of 10.0 households per ha (bottom). The table shows the network cost in CHF on the left and the corresponding network lengths and number of branches on the right.

Low Household Density										
Scenario	Cost in CHF					Length in km (no. of branches)				
	<i>N</i>	20	40	60	80	100	20	40	60	80
Base Scenario	296,861	921,418	2,109,621	4,079,893	6,570,501	4.869 (2.9)	12.849 (6.1)	27.059 (10.9)	49.845 (17.8)	82.445 (26.6)
Increased Coincidence	308,439	960,924	2,161,832	4,084,742	6,573,492	5.093 (3.5)	13.467 (6.9)	27.901 (11.9)	50.297 (18.8)	82.183 (27.2)
Worst Case: Charging of EVs	420,528	1,524,283	3,510,802	6,069,448	8,852,005	6.284 (5.3)	20.210 (12.5)	48.188 (23.9)	76.980 (34.0)	116.451 (49.4)
Worst Case: PV Feed-In	313,392	950,867	2,157,650	4,151,948	6,580,144	5.071 (3.4)	13.278 (6.7)	27.95 (11.9)	50.943 (19.1)	82.452 (27.7)

High Household Density										
Scenario	Cost in CHF					Length in km (no. of branches)				
	<i>N</i>	20	40	60	80	100	20	40	60	80
Base Scenario	17,158	37,044	60,879	84,778	111,701	0.439 (2.2)	0.900 (2.4)	1.404 (2.7)	1.879 (3.1)	2.392 (3.4)
Increased Coincidence	17,395	39,192	66,380	94,680	127,253	0.440 (2.3)	0.938 (3.2)	1.517 (4.5)	2.064 (5.2)	2.587 (5.6)
Worst Case: Charging of EVs	20,088	51,536	89,180	138,822	180,610	0.483 (3.4)	1.120 (5.7)	1.963 (8.9)	2.514 (9.5)	3.184 (10.5)
Worst Case: PV Feed-In	17,452	39,086	66,622	94,685	127,250	0.440 (2.2)	0.936 (3.2)	1.516 (4.4)	2.061 (5.2)	2.597 (5.6)

Table IV-5: Top: Low household density. Bottom: High household density. Left: Comparison of network cost in CHF for various number of loads *N*. Reported is the cost averaged across 50 instances. The cells are shaded based on average cost. Right: Comparison of network length in km with number of branches in parenthesis. The cells are shaded based on average length.

First, an overall observation can be made when comparing the main case from Section 4.2.1 (medium household density) to the two other cases (low and high household density). While overall patterns are similar, the magnitude of the effect caused by changes in the coincidence factor are higher in the main case than in both other cases. Thus, when starting at the main case, both increasing and decreasing the household density mitigates the effect of load coincidence. Possible reasons for this are discussed below.

Let's start by comparing the network cost. Overall, the costs for the case with low household density are much higher compared to the other cases. This is to be expected because this case has 10- or 100-times fewer loads per ha. This causes the grid lines to be much longer and, because of the higher voltage drops along the longer lines, this causes the capacities to be higher as well. Both factors contribute to higher network cost. Further, similar patterns as in the main case are observed: The network cost grows as the networks get larger, the base scenario has the lowest cost and the worst case scenario for simultaneous charging of EVs has the highest cost. The other two scenarios (increased coincidence and worst case: PV feed-in) are fairly comparable in cost. For the case with high household density, the relative cost gap between the base scenario and the other three scenarios increases with  $N$ : The gap increases from 1–17% ( $N = 20$ ) to 14–62% ( $N = 100$ ). For the case with low household density, the gap decreases from 4–42% ( $N = 20$ ) to 0–35% ( $N = 100$ ). The reason for this is related to the changes in network length and branching and is detailed below.

Regarding network length and branching, both figures increase with  $N$  as before. However, in stark contrast to before, all networks except the ones from the worst case scenario for charging of EVs are fairly similar in length and branching. This can be explained as follows. In the high density case, distances and therefore voltage drops are relatively short. Therefore, for almost all cases, a network layout close to the MST can be achieved without violating line sizing and voltage drop constraints. The MST is the optimal layout for situations where voltage drops can be neglected (Remark 2). For the low density case, we have the opposite situation: Distances are relatively long, and therefore, the optimal layout is close to the starred network (Remark 3). This is seen in the data by the fact that each branch contains on average only 2–6 loads. The cost function has been chosen to resemble real-world costs for the construction of electricity distribution networks. The construction cost component is much higher than the material cost component. Therefore, even if a higher capacity

must be chosen for one of the scenarios when compared to the base scenario, the cost difference will not be as high.

Regarding runtimes, Table IV-6 shows the runtimes for both cases. The highest runtimes observed are 6.0 hours (21,613 seconds) per network instance. This is sufficient for real-world applications. Similar observations than in Section 4.2.1 can be made, in particular, that the smaller the branching, the higher the runtimes. This holds true when comparing both between the cases (the case with high household density has higher runtimes than the other cases) and within the cases (the base case consistently exhibits the highest runtimes compared to all other scenarios).

<b>Low Household Density</b>						
	Time in seconds					
<b>Scenario</b>	<b><i>N</i></b>	<b>20</b>	<b>40</b>	<b>60</b>	<b>80</b>	<b>100</b>
Base Scenario		13.83	77.81	193.65	312.85	799.85
Increased Coincidence		11.05	60.47	170.57	346.78	754.07
Worst Case: Charging of EVs		5.48	30.35	89.79	160.20	589.78
Worst Case: PV Feed-In		10.46	58.93	159.90	300.60	761.43

<b>High Household Density</b>						
	Time in seconds					
<b>Scenario</b>	<b><i>N</i></b>	<b>20</b>	<b>40</b>	<b>60</b>	<b>80</b>	<b>100</b>
Base Scenario		15.97	658.06	4,181.01	10,578.89	21,613.37
Increased Coincidence		24.70	368.71	1,398.35	3,759.04	6,823.30
Worst Case: Charging of EVs		16.75	171.30	641.68	1,480.75	1,906.14
Worst Case: PV Feed-In		24.14	357.31	1,389.21	3,569.09	6,802.02

Table IV-6: Comparison of runtimes in seconds for various number of loads  $N$ . Reported is the averaged runtime across 50 instances.

In summary, in the cases with low or high household density, changes in load coincidence have a smaller effect on network cost, length and branching, when compared to the main case. Nevertheless, network cost can be up to 62% higher compared to the base scenario. Simultaneous charging of EVs has the most severe effect on network cost. The cost effect of the scenario of increased coincidence is comparable to the effect that is expected from an uncontrolled PV feed-in.

## 5 Real-World Experiments

In this section, the effects of load coincidence on network cost and layouts are evaluated on the same set of low voltage distribution networks as in Section 4.

### 5.1 Experimental Setup

The locations for the real-world experiments are based on the identical sample of 74 low voltage distribution networks than the one in Section 4. Each set of locations entails one transformer. The number of loads per network in the sample ranges from 12 to 68, with an average of 36.7 loads and a median of 32 loads. These loads correspond to the vertices of the network. The costs for the networks range from CHF 33,500 for the cheapest network to CHF 1.7 million for the most expensive. The average network cost is CHF 251,400. The networks sum up to a combined value of CHF 18.6 million. This cost only includes material and construction and excludes planning and overhead costs. On average, each network covers an area of 35.1 ha, with a median size of 20.6 ha per network. Therefore, the average household density is 1.05 households per ha. Data preparation and pre-processing is detailed in Section 4.1.

Similar to Section 4, the networks are grouped by the number of loads  $N$  contained in each network. The groups of networks range from  $N \in [10, 19]$  to  $N \in [60, 69]$ . The number of networks in each group is reported alongside the results.

### 5.2 Results

Table IV-7 displays the results for real-world experiments. The table shows the network cost in CHF on the left and the corresponding network lengths and number of branches on the right. Below, the findings on cost, network length and branching are discussed.

Scenario	N	Cost in CHF						Length in km (no. of branches)					
		[10, 19]	[20, 29]	[30, 39]	[40, 49]	[50, 59]	[60, 69]	[10, 19]	[20, 29]	[30, 39]	[40, 49]	[50, 59]	[60, 69]
Base Scenario		58,659	102,732	68,831	140,317	82,988	313,230	1.290 (2.1)	2.017 (1.9)	1.409 (1.3)	2.806 (2.3)	1.750 (1.9)	6.373 (8.2)
Increased Coincidence		73,431	170,058	144,000	258,160	200,052	440,991	1.541 (3.1)	3.467 (6.0)	3.020 (6.9)	5.335 (9.3)	4.484 (10.6)	8.490 (13.9)
Worst Case: Charging of EVs		82,350	177,729	164,927	265,218	215,318	493,987	1.713 (3.6)	3.57 (6.6)	3.388 (7.7)	5.441 (9.7)	4.776 (12.4)	9.150 (15.4)
Worst Case: PV Feed-In		60,051	113,821	110,710	196,201	172,563	375,849	1.299 (1.9)	2.272 (3.1)	2.401 (5.3)	4.172 (8.1)	4.032 (9.4)	7.622 (12.1)
Number of networks in sample group		14	19	9	9	14	9	14	19	9	9	14	9

Table IV-7: Left: Comparison of network cost for networks with various number of loads  $N$ . Reported is the average cost. The cells are shaded based on average cost. Right: Comparison of network length in km with number of branches in parenthesis. The cells are shaded based on average length. After a maximum calculation time of 24 hours per network, the cheapest network is returned.

Regarding network cost, the observed costs are comparable in order of magnitude to those in the main scenario in Section 4.2.1. It can be observed that, in general, network costs grow as the networks get larger. However, because of the smaller sample size and higher homogeneities among the networks when compared to the computational experiments in Section 4, there are situations, where networks with fewer households exhibit larger costs than those with more households. Similar to the computational experiments, the base scenario has lowest cost for all network sizes, while the worst case scenario for simultaneous charging of EVs has the highest cost. However, other than in the computational experiments, in the real-world case study, the two remaining scenarios (increased coincidence and worst case: PV feed-in) are further apart from each other in terms of cost: The networks in the worst case scenario for PV feed-in have significantly lower cost than the networks in the increased coincidence scenario, as is expected. When looking at the relative cost gap between the base scenario and the other three scenarios increases, we observe a gap of 25–109 % for the increased coincidence scenario, a gap of 40–159 % for the worst case scenario for simultaneous charging of EVs, and a gap of 2–108 % for the worst case scenario for PV feed-in.

Regarding network length and branching, in general, both figures increase with  $N$ . Similar to the network cost, the four scenarios are much more distinct from each other than in the computational experiments: Lowest lengths and branching are observed for the base scenario, followed by the worst case scenario for PV feed-in, the increased coincidence scenario, and the worst case scenario for simultaneous charging of EVs. The networks from the increased coincidence scenario are 19–156 %

longer compared to the base scenario, the networks from the worst case scenario for simultaneous charging of EVs are 33–173 % longer, the networks from the worst case scenario for PV feed-in are 1–130 % longer.

In short, the real-world case study confirms the findings from the computational experiments. In all three scenarios, network costs can more than double. Simultaneous charging of EVs potentially has the most severe effect on network cost, followed by the scenario of increased coincidence, followed by the scenario for an uncontrolled PV feed-in.

## 6 Summary

The computational experiments clearly show that the change in load coincidence on its own has a significant effect on network cost and layout. This is confirmed in the real-world case study, where results show that increasing load coincidence can lead to a doubling of network cost. Particularly, the uncontrolled simultaneous charging of EVs poses stress on the networks, yielding much higher cost. This has severe implications for both distribution network operators and regulators, as is discussed in the last chapter of this thesis. Another interesting observation is that, in the computational experiments, three cases were compared: one with low, one with medium and one with high household density. The largest effects of load coincidence on network cost could be observed in the case with medium household density. This implies that future changes in load coincidence could cause the highest complications for network operators in suburban setting, rather than very sparsely populated rural settings, or highly dense urban settings.

## CONCLUDING REMARKS

### 1 Summary of Findings

At the beginning of this thesis, we saw how changes in the demand patterns of electricity use is causing problems in today's electricity networks. The rolling power shut-offs in California exemplified the importance of load coincidence in the context of designing and operating distribution networks. We also saw that in the future, load coincidence is expected to increase due to trends in the electricity sector, namely (A) direct adaptations to climate change, (B) an increasing share of EVs, and (C) a further decentralized energy generation.

This motivated creating the capacitated arborescence with voltage drops and load coincidence problem, a decision problem for the design of distribution networks. The novelties of this problem are that it treats voltage drops in a more realistic way than any other related network design problem and that it accounts for the way that loads coincide. This brings two advantages. First, it allows for more cost-efficient network design (Chapter III). Second, it allows to analyze the effect of load coincidence on future network cost and layouts (Chapter IV).

In Chapter III, it has been shown that the CAVLP is NP-hard and contains complex nonlinearities. This inherent complexity of the problem prohibits exact solutions even for smallest instances. This lead to the development of heuristical solution methods,

which make it possible to solve the problem in a reasonable amount of time even for larger instances. These solution approaches have been tested using simulated problem instances and real-world networks from a Swiss electricity company. The computational experiments and the real-world case study demonstrated that the solution approaches are computationally tractable. Further, they point towards significant cost savings when using the CAVLP (and the corresponding solution methods) compared to traditional design approaches: In the case study, the networks created using the proposed solution methods showed relative cost savings of over 39% compared to the real-world networks designed using conventional techniques. The absolute cost savings for larger networks were as high as CHF 0.26 million per network. This demonstrated the practical value-add that this thesis can provide to practitioners.

In Chapter IV, the effect of load coincidence on network cost and layouts was investigated. The question at hand was, how future demand scenarios influence network design. Specifically, three scenarios were tested against a base scenario representing the status quo: (A) increased coincidence, (B) worst case scenario for simultaneous charging of EVs, and (C) worst case scenario for PV feed-in. These three scenarios correspond to the trends (A)–(C) identified above. The results show a significant effect of all three scenarios on both network cost and layout, with (B) the worst case scenario for simultaneous charging of EVs having the largest effect: In the computational experiments, networks for this scenario were up to 84% more expensive compared to the base scenario. In the real-world case study, the difference was even higher (173% higher cost compared to base scenario for the largest problem instances). The experiments also show that the largest effects from load coincidence can be expected in regions with medium household density, meaning that the issue might not be as severe in highly-populated urban areas or sparsely-populated rural areas when compared to suburban regions.

The above observations have several implications for decision makers, especially for distribution system operators and regulators. These implications are discussed below.

## 2 Implications

The experiments clearly showcase that the effects of load coincidence can be severe and cannot be neglected by any stakeholder, particularly by DSOs (who are ultimately responsible for a reliable and efficient energy supply to the consumers) and regulators (who are responsible for providing a framework for a reliable and efficient operation of electricity infrastructure). An overview of implications for DSOs and regulators is shown in Figure V-1. This list is non-exhaustive.

The scenario with the largest effect on network cost and layout was (B) worst case scenario for simultaneous charging of EVs. The figures shown in this thesis correspond to a worst case scenario. Because of the relatively small number of EVs on the streets right now, this is not much affecting network operations today. However, the number of EVs is projected to grow rapidly in the future (see Chapter I). This implies that DSOs need to carefully monitor network stability as the number of EVs increases. Potential reinforcements to the network infrastructure might be inevitable. Electricity companies, however, might also think about methods to gain more control over the charging behavior of their customers. This could be done by actively controlling the

	Implications for DSOs	Implications for regulators
<b>A</b> Direct adaptations to climate change	<ul style="list-style-type: none"> <li>Perform rolling shutoffs in emergency situations</li> <li>Reinforce distribution infrastructure if necessary</li> </ul>	<ul style="list-style-type: none"> <li>Create awareness of the problem and influence people's behavior</li> <li>Promote/subsidize more energy-efficient cooling systems and insulation</li> <li>Increase all efforts to combat climate change</li> </ul>
<b>B</b> Increasing share of EVs	<ul style="list-style-type: none"> <li>Carefully monitor network stability as the number of EVs increases</li> <li>Reinforce distribution infrastructure if necessary</li> <li>Gain more control over the charging behavior of their customers (e.g., by actively controlling the charging process or by tariffs that incentivize off-peak charging)</li> </ul>	<ul style="list-style-type: none"> <li>Create regulatory framework to support DSOs in their effort</li> <li>Promote/subsidize alternative alternative technologies (e.g., hydrogen EVs)</li> </ul>
<b>C</b> Further decentralized energy generation	<ul style="list-style-type: none"> <li>Promote batteries and other storage technologies</li> </ul>	<ul style="list-style-type: none"> <li>Promote/subsidize batteries and other storage technologies</li> <li>Adjust laws regarding electricity feed-in (e.g., reduce feed-in tariffs)</li> </ul>

Figure V-1: Implications for DSOs and regulators. The implications for DSOs and regulators can be broken down by the three trends (A)–(C). The list is non-exhaustive.

charging process or by offering flexible tariffs that incentivize off-peak charging. Regulators might encourage such initiatives (e. g., by creating the regulatory frameworks that enable specific tariffs). Regulators might also think about promoting or subsidizing alternative technologies to battery EVs, e. g., hydrogen EVs.

Regarding the implications of scenario (A), we already saw reactions from DSOs and regulators in the introduction of this thesis. In the short term, rolling shutoffs and reinforcements to the distribution network will most likely be necessary. Mid- to long-term, regulators should promote more energy-efficient cooling systems and insulation in order to mitigate the direct effects of climate change. This should happen in addition to creating awareness for the problem and influencing people's behavior (as we have seen from the Governor of California urging his people to save energy during peak hours). Of course, moving towards a greener, carbon-neutral society as a whole can be seen as a response to scenario (A). The sooner this goal can be achieved, the less severe the direct effects of climate change will be. Therefore, increasing all efforts to combat climate change are a direct implication of scenario (A).

Regarding the implications of scenario (C), ways to prevent an overload of the distribution network through electricity feed-in have to be found. Batteries and other energy storage technologies seem very promising. They provide benefits both to consumers (by making them more energy-independent and allowing them to have access to cheaper electricity) and system operators (by mitigating excess feed-in and thus provide relief to the network infrastructure). Regulators can also adjust the laws regarding feed-in (e. g., by reducing feed-in tariffs) and promote battery storage technologies (e. g., by investing in research and development).

### 3 Outlook

Finally, there are potential research topics which could be pursued in the future. There are several additional applications where the approach presented in this thesis can provide value. Below, four potential research topics are briefly explored. These topics are depicted in Figure V-2.

First, regarding the model, one avenue for further research is to integrate additional network technologies other than grid lines into the decision problem. For instance, the integration of storage technologies could mitigate the effect of coinciding demands. The same holds true for the integration of decentralized generation. Note that the integration of PV feed-in as it is done in Chapter IV represents a worst case scenario. As it is done currently, the model only supports worst case feed-in or worst case consumption, not a combination of both. Eventually, integration of distributed

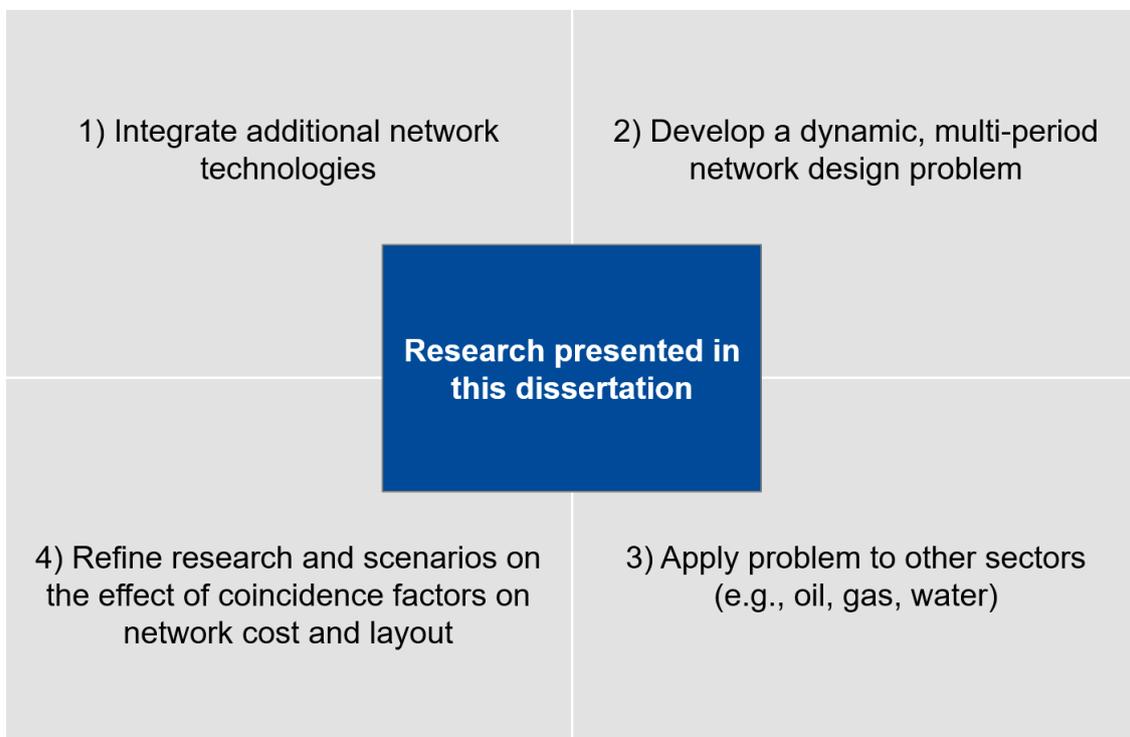


Figure V-2: Four future research topics can be explored: 1) integrate additional technologies other than grid lines (e.g., on-load tap changers, storage, decentralized generation), 2) develop a dynamic, multi-period network design problem (or modify the problem to allow expansion planning), 3) apply the problem to other sectors with similar physical constraints (e.g., to oil, gas, or water networks), 4) refine research and scenarios on the effect of coincidence factors on network cost and layout (e.g., by combining a change in coincidence factor with an actual change in the peak loads).

generation (e. g., PV systems) with storage can help reducing network investment cost because it mitigates the effect of coinciding peak loads. Second, future research could also integrate the CAVLP into a dynamic, multi-period distribution network design problem. Slight modifications to the model could also be made to allow for network expansion planning instead of greenfield network design. This would expand the application of the model to existing networks rather than ones that are about to be built. Third, the proposed methods can also facilitate the solution of related optimization problems with similar physical constraints in other sectors. Examples are network design problems with pressure drops, such as in oil, gas, or water networks, as well as reliability constraints in transportation networks. Studies on precipitation and flooding could also benefit from a model that accounts for coinciding demand. The (non-)coinciding demand in the model presented then translates into (non-)coinciding water supply. Fourth and finally, the research on the effect of coincidence factors on network cost and layouts can be expanded to include more or refined scenarios. This effort could be supported by the increasing availability of load data due to smart meter technologies (see Chapter II). Another possibility to create additional scenarios could be to combine changes in the coincidence factor with changes in the peak loads.

In conclusion, there are several open research questions which could be answered in the future. The optimization problem presented in this thesis and the corresponding solution methods can provide valuable tools to many stakeholders in the electricity sector and beyond.

## CHAPTER VI

### BIBLIOGRAPHY

- Adams, R. N., and Laughton, M. A. 1974. "Optimal planning of power networks using mixed-integer programming. Part 1: Static and time-phased network synthesis," *Proc Inst Electr Eng* (121:2), pp. 139–147.
- André, J., Auray, S., Brac, J., de Wolf, D., Maisonnier, G., Ould-Sidi, M.-M., and Simonnet, A. 2013. "Design and dimensioning of hydrogen transmission pipeline networks," *Eur J Oper Res* (229:1), pp. 239–251.
- Avella, P., Villacci, D., and Sforza, A. 2005. "A Steiner arborescence model for the feeder reconfiguration in electric distribution networks," *Eur J Oper Res* (164:2), pp. 505–509.
- Balakrishnan, A., Li, G., and Mirchandani, P. 2017. "Optimal Network Design with End-to-End Service Requirements," *Oper Res* (65:3), pp. 729–750.
- Beaty, H. W., and Fink, D. G. (eds.) 2013. *Standard handbook for electrical engineers*, New York, NY: McGraw-Hill, 16th ed.
- Boait, P., Advani, V., and Gammon, R. 2015. "Estimation of demand diversity and daily demand profile for off-grid electrification in developing countries," (29), pp. 135–141.
- Boulaxis, N. G., and Papadopoulos, M. P. 2002. "Optimal feeder routing in distribution system planning using dynamic programming technique and GIS facilities," *IEEE Trans Power Delivery* (17:1), pp. 242–247.
- Brimberg, J., Hansen, P., Lin, K.-W., Mladenović, N., and Breton, M. 2003. "An Oil Pipeline Design Problem," *Oper Res* (51:2), pp. 228–239.
- Bruno, G., and Laporte, G. 2002. "A Simple Enhancement of the Esau-Williams Heuristic for the Capacitated Minimum Spanning Tree Problem," *The Journal of the Operational Research Society* (53:5), pp. 583–586.
- Bucher, D. B., Kufner, C. L., Schlueter, A., Carell, T., and Zinth, W. 2016. "UV-induced charge transfer states in DNA promote sequence selective self-repair," *Journal of the American Chemical Society* (138:1), pp. 186–190.

- Bucher, D. B., Schlueter, A., Carell, T., and Zinth, W. 2014. "Watson–Crick Base Pairing Controls Excited–State Decay in Natural DNA," *Angewandte Chemie International Edition* (53:42), pp. 11,366–11,369.
- Burroughs, N. B. 2020. "Lights Dim and Worries Mount as a Heat Wave Roasts California," *New York Times* .
- Cayley, A. 1889. "A theorem on trees," *Quart J Pure Appl Math* (23), pp. 376–378.
- CENELEC 2010. "Voltage characteristics of electricity supplied by public electricity networks," .
- Chandy, K. M., and Lo, T. 1973. "The capacitated minimum spanning tree," *Networks* (3:2), pp. 173–181.
- Chen, L., Singh, V. P., Shenglian, G., Hao, Z., and Li, T. 2012. "Flood Coincidence Risk Analysis Using Multivariate Copula Functions," *J Hydrol Eng* (17:6), pp. 742–755.
- Cossi, A. M., Romero, R., and Mantovani, J. 2005. "Planning of Secondary Distribution Circuits Through Evolutionary Algorithms," *IEEE Trans Power Delivery* (20:1), pp. 205–213.
- Dahl, G., Gouveia, L., and Requejo, C. 2006. "On Formulations and Methods for the Hop-Constrained Minimum Spanning Tree Problem," in *Handbook of Optimization in Telecommunications*, M. G. C. Resende and P. M. Pardalos (eds.), Boston, MA: Springer, pp. 493–515.
- Davis, L. W., and Gertler, P. J. 2015. "Contribution of air conditioning adoption to future energy use under global warming," *Proceedings of the National Academy of Sciences of the United States of America* (112:19), pp. 5962–5967.
- de Boeck, J., and Fortz, B. 2018. "Extended formulation for hop constrained distribution network configuration problems," *Eur J Oper Res* (265:2), pp. 488–502.
- Deschênes, O., and Greenstone, M. 2011. "Climate Change, Mortality, and Adaptation: Evidence from Annual Fluctuations in Weather in the US," *American Economic Journal: Applied Economics* (3:4), pp. 152–185.
- Dickert, J., and Schegner, P. 2010. "Residential load models for network planning purposes," *Proceedings - International Symposium: Modern Electr Power Syst, MEPS'10* .
- Domingo, C. M., Gomez, T. S. R., Sanchez-Miralles, A., Peco, J. P. G., and Candela, A. M. 2011. "A Reference Network Model for Large-Scale Distribution Planning With Automatic Street Map Generation," *IEEE Transactions on Power Systems* (26:1), pp. 190–197.
- Esau, L. R., and Williams, K. C. 1966. "On teleprocessing system design, Part II: A method for approximating the optimal network," *IBM Systems Journal* (5:3), pp. 142–147.
- European Commission 2019. "The European Green Deal: COM(2019) 640," .

- Falaghi, H., Singh, C., Haghifam, M.-R., and Ramezani, M. 2011. "DG integrated multistage distribution system expansion planning," *Int J Electr Power Energy Syst* (33:8), pp. 1489–1497.
- Fisher, E. B., O'Neill, R. P., and Ferris, M. C. 2008. "Optimal Transmission Switching," *IEEE Trans Power Syst* (23:3), pp. 1346–1355.
- Freitas, K. B., Toledo, C. F. M., and Delbem, A. C. B. 2016. "Optimal reconfiguration of electric power distribution systems using exact approach," in *2016 12th IEEE International Conference on Industry Applications (INDUSCON)*, , IEEE.
- Gan, C. K., Mancarella, P., Pudjianto, D., and Strbac, G. 2011. "Statistical appraisal of economic design strategies of LV distribution networks," *Electr Power Syst Res* (81:7), pp. 1363–1372.
- Gaul, A., Czajkowski, C., Voit, S., and Übermasser, S. 2017. "Rollout e-mobility – the next big challenge for network operations and network planning," in *Grid Integration of Electric Mobility*, J. Liebl (ed.), Proceedings, Wiesbaden: Springer Fachmedien Wiesbaden and Imprint: Springer Vieweg.
- Gendreau, M., and Potvin, J.-Y. 2014. "Tabu Search," in *Search Methodologies*, E. K. Burke and G. Kendall (eds.), Boston, MA: Springer, pp. 243–263.
- Georgilakis, P. S., and Hatziaargyriou, N. D. 2015. "A review of power distribution planning in the modern power systems era: Models, methods and future research," *Electr Power Syst Res* (121), pp. 89–100.
- Glamocanin, V. 1990. "Optimal loss reduction of distributed networks," *IEEE Trans Power Syst* (5:3), pp. 774–782.
- Glover, F. 1989. "Tabu Search—Part I," *ORSA J on Comp* (1:3), pp. 190–206.
- Glover, J. D., Sarma, M. S., and Overbye, T. J. 2012. *Power system analysis and design*, Cengage Learning, 5th ed.
- Gonen, T. 2015. *Electric Power Distribution Engineering*, CRC Press.
- Gouveia, L., and Magnanti, T. L. 2003. "Network flow models for designing diameter-constrained minimum-spanning and Steiner trees," *Networks* (41:3), pp. 159–173.
- Gurobi Optimization 2017. "Gurobi Optimizer," .
- Gust, G., Neumann, D., Flath, C. M., Brandt, T., and Ströhle, P. 2017. "How a traditional company seeded new analytics capabilities," *MIS Quart Exec* (16:3), pp. 215–230.
- Gwisdorf, B., Stepanescu, S., and Rehtanz, C. 2010. "Effects of Demand Side Management on the planning and operation of distribution grids," in *2010 IEEE PES Innovative Smart Grid Technologies Conference Europe*, I. Staff (ed.), Gothenburg: I E E E.

- Hansen, P., and Mladenović, N. 2014. "Variable Neighborhood Search," in *Search Methodologies*, E. K. Burke and G. Kendall (eds.), Boston, MA: Springer US, pp. 313–337.
- Hansen, P., Mladenović, N., Brimberg, J., and Pérez, J. A. M. 2019. "Variable neighborhood search," in *Handbook of metaheuristics*, Cham: Springer, pp. 57–97.
- Herman, R., and Kritzing, J. J. 1993. "The statistical description of grouped domestic electrical load currents," (27:1), pp. 43–48.
- IEA 2019. *Market Report Series: Renewables 2019: Analysis and Forecasts to 2024*.
- IEA 2020. *Global EV Outlook 2020: Entering the Decade of Electric Drive*.
- IEC 2015a. "Electromagnetic compatibility (EMC): Part 4-30: Testing and measurement techniques - Power quality measurement methods," .
- IEC 2015b. "Strategic asset management of power networks: White Paper," .
- Jabr, R. A., Singh, R., and Pal, B. C. 2012. "Minimum Loss Network Reconfiguration Using Mixed-Integer Convex Programming," *IEEE Trans Power Syst* (27:2), pp. 1106–1115.
- Jenabi, M., Fatemi Ghomi, S. M. T., Torabi, S. A., and Hosseinian, S. H. 2015. "Acceleration strategies of Benders decomposition for the security constraints power system expansion planning," *Annals of Oper Res* (235:1), pp. 337–369.
- Jothi, R., and Raghavachari, B. 2004. "Revisiting Esau-Williams' algorithm: on the design of local access networks," in *IN PROC. 7TH INFORMS TELECOMMUNICATIONS CONF*, .
- Kaur, D., and Sharma, J. 2008. "Optimal conductor sizing in radial distribution systems planning," *International Journal of Electrical Power & Energy Systems* (30:4), pp. 261–271.
- Khodaei, A., Shahidehpour, M., and Kamalinia, S. 2010. "Transmission Switching in Expansion Planning," *IEEE Trans Power Syst* (25:3), pp. 1722–1733.
- Kocuk, B., Jeon, H., Dey, S. S., Linderoth, J., Luedtke, J., and Sun, X. A. 2016. "A Cycle-Based Formulation and Valid Inequalities for DC Power Transmission Problems with Switching," *Oper Res* (64:4), pp. 922–938.
- Kong, T., Cheng, H., Hu, Z., and Yao, L. 2009. "Multiobjective planning of open-loop MV distribution networks using ComGIS network analysis and MOGA," *Electr Power Syst Res* (79:2), pp. 390–398.
- Konstantelos, I., Sun, M., and Strbac, G. 2014. *Quantifying demand diversity of households*.
- Märkle-Huß, J., Feuerriegel, S., and Neumann, D. 2019. "Cost minimization of large-scale infrastructure for electricity generation and transmission," *Omega* p. 102071.

- Mathews, J. A., and Tan, H. 2014. "Economics: Manufacture renewables to build energy security," *Nature News* (513:7517), p. 166.
- Mladenović, N., and Hansen, P. 1997. "Variable neighborhood search," *Comput & Oper Res* (24:11), pp. 1097–1100.
- Molzahn, D. K., Dörfler, F., Sandberg, H., Low, S. H., Chakrabarti, S., Baldick, R., and Lavaei, J. 2017. "A Survey of Distributed Optimization and Control Algorithms for Electric Power Systems," *IEEE Transactions on Smart Grid* (8:6), pp. 2941–2962.
- Nahman, J. M., and Peric, D. M. 2008. "Optimal Planning of Radial Distribution Networks by Simulated Annealing Technique," *IEEE Trans Power Syst* (23:2), pp. 790–795.
- Navarro, A., and Rudnick, H. 2009. "Large-Scale Distribution Planning—Part I: Simultaneous Network and Transformer Optimization," *IEEE Trans Power Syst* (24:2), pp. 744–751.
- Newsom, G. 2020. "As West Coast Faces Historic Heat Wave & Energy Shortages, Governor Newsom Signs Heat Emergency Proclamation to Free Up Energy Capacity," .
- Nickel, D., and Braunstein, H. 1981. "Distribution transformer loss evaluation: II-Load characteristics and system cost parameters," *IEEE Transactions on Power Apparatus and Systems* (PAS-100:2), pp. 798–811.
- Papadimitriou, C. H. 1978. "The complexity of the capacitated tree problem," *Networks* (8:3), pp. 217–230.
- Parada, V., Ferland, J. A., Arias, M., Schwarzenberg, P., and Vargas, L. 2010. "Heuristic Determination of Distribution Trees," *IEEE Trans Power Delivery* (25:2), pp. 861–869.
- Parshall, L., Pillai, D., Mohan, S., Sanoh, A., and Modi, V. 2009. "National electricity planning in settings with low pre-existing grid coverage: Development of a spatial model and case study of Kenya," (37:6), pp. 2395–2410.
- Pineda, S., and Morales, J. M. 2016. "Capacity expansion of stochastic power generation under two-stage electricity markets," *Comput & Oper Res* (70), pp. 101–114.
- Prim, R. C. 1957. "Shortest Connection Networks And Some Generalizations," *Bell Syst Tech J* (36:6), pp. 1389–1401.
- Qi, W., Liang, Y., and Shen, Z.-J. M. 2015. "Joint Planning of Energy Storage and Transmission for Wind Energy Generation," *Oper Res* (63:6), pp. 1280–1293.
- Requejo, C., and Santos, E. 2009. "Greedy heuristics for the diameter-constrained minimum spanning tree problem," *J of Math Sci* (161:6), pp. 930–943.
- Resch, M., Bühler, J., Klausen, M., and Sumper, A. 2017. "Impact of operation strategies of large scale battery systems on distribution grid planning in Germany," (74), pp. 1042–1063.

- Richardson, P., Flynn, D., and Keane, A. 2010. "Impact assessment of varying penetrations of electric vehicles on low voltage distribution systems," in *2010 IEEE Power and Energy Society General Meeting*, I. Staff (ed.).
- Richardson, P., Flynn, D., and Keane, A. 2012. "Optimal Charging of Electric Vehicles in Low-Voltage Distribution Systems," *IEEE Trans Power Syst* (27:1), pp. 268–279.
- Rossi, A., Aubry, A., and Jacomino, M. 2012. "Connectivity-and-hop-constrained design of electricity distribution networks," *Eur J Oper Res* (218:1), pp. 48–57.
- Rusck, S. 1956. "The simultaneous demand in distribution network supplying domestic consumers," *ASEA Journal* (10:11), pp. 59–61.
- Sauhats, A., Petrichenko, L., Beryozkina, S., and Jankovskis, N. 2016. "Stochastic planning of distribution lines," in *13th International Conference on the European Energy Market (EEM)*, , pp. 1–5.
- Schlueter, A., Gust, G., Feuerriegel, S., and Neumann, D. 2017. "Minimization Of Investment Costs For Designing Electricity Distribution Networks," 2017 INFORMS Annual Meeting – Houston, Texas October 22-25.
- Schwab, A. J. 2017. *Elektroenergiesysteme: Erzeugung, Übertragung und Verteilung elektrischer Energie*, Springer-Verlag.
- Schweitzer, E., Saha, S., Scaglione, S., Johnson, N. G., and Arnold, D. 2020. "Lossy DistFlow Formulation for Single and Multiphase Radial Feeders," *IEEE Transactions on Power Systems* (35:3), pp. 1758–1768.
- Singh, K. J., and Mason, A. 2004. "Column-generation for capacity-expansion planning of electricity distribution networks," in *Proceedings of the 39th Annual OR Society of New Zealand Conference*, .
- Singh, K. J., Philpott, A. B., and Wood, R. K. 2009. "Dantzig-Wolfe Decomposition for Solving Multistage Stochastic Capacity-Planning Problems," *Oper Res* (57:5), pp. 1271–1286.
- Steele, J. M., Shepp, L. A., and Eddy, W. F. 1987. "On the number of leaves of a Euclidean minimal spanning tree," *J Appl Probab* (24:04), pp. 809–826.
- Toth, P., and Vigo, D. 1995. "An exact algorithm for the capacitated shortest spanning arborescence," *Annals of Oper Res* (61:1), pp. 121–141.
- UN 2015. "Paris Agreement," United Nations Framework Convention on Climate Change.
- UN 2018. *Global trends in renewable energy investment 2018*.
- U.S. Department of Energy 2014. "Infographic: Understanding the Grid," .
- U.S. Department of Energy 2018. "Major utilities continue to increase spending on U.S. electric distribution systems," .

- U.S. Department of Energy 2020. "Annual Energy Outlook 2020 with projections to 2050," .
- U.S. House of Representatives 2019. "Recognizing the duty of the Federal Government to create a Green New Deal (H. Res. 109)," .
- Vallés, M., Reneses, J., Frías, P., and Mateo, C. 2016. "Economic benefits of integrating Active Demand in distribution network planning: A Spanish case study," (136), pp. 331–340.
- Villumsen, J. C., and Philpott, A. B. 2012. "Investment in electricity networks with transmission switching," *Eur J Oper Res* (222:2), pp. 377–385.
- Voß, S. 2001. *Capacitated Minimum Spanning Trees*, Boston, MA: Springer US, pp. 225–235.
- Weedy, B. B. 2012. *Electric power systems*, Chichester, West Sussex, UK: John Wiley & Sons Ltd, 5th ed.
- Widén, J., and Wäckelgård, E. 2010. "A high-resolution stochastic model of domestic activity patterns and electricity demand," (87:6), pp. 1880–1892.
- Wieland, T., Reiter, M., Schmutzner, E., and Fickert, L. (eds.) 2015. *Modern Grid Planing: A Probabilistic Approach for Low Voltage Networks facing New Challenges*.
- Yan, B., and Chen, L. 2013. "Coincidence probability of precipitation for the middle route of South-to-North water transfer project in China," *J Hydrol* (499), pp. 19–26.
- Zohrizadeh, F., Josz, C., Jin, M., Madani, R., Lavaei, J., and Sojoudi, S. 2020. "A survey on conic relaxations of optimal power flow problem," *Eur J Oper Res* (287:2), pp. 391–409.



# CURRICULUM VITAE

## Personal Information

---

DATE OF BIRTH: | February 10, 1989 in Backnang, Germany

MAIL: | alexander.schlueter@is.uni-freiburg.de

## Education

---

09/2011 – 04/2014: **Ludwig-Maximilians-Universität Munich** – Master of Science (Physics)  
Thesis: “Excited State Dynamics of Nucleobases in Natural DNA  
Investigated by Femtosecond Infrared Spectroscopy“

09/2011 – 03/2012: **University of British Columbia Vancouver, Canada** – Studies abroad

10/2008 – 08/2011: **Ludwig-Maximilians-Universität Munich** – Bachelor of Science (Physics)

06/2008: **Georg Büchner Gymnasium Winnenden** – Abitur

## Work Experience

---

Since 01/2015: **McKinsey & Company** — Consultant

05/2014 – 07/2014: **McKinsey & Company** — Intern

2010 – 2013: **Ludwig-Maximilians-Universität Munich** -- Teaching assistant

## Awards and Scholarships

---

2010 – 2014: Scholarship of the **German National Academic Foundation**  
(**Studienstiftung des deutschen Volkes**)

2011: Scholarship of the **Ludwig-Maximilians-Universität Munich**  
to study at the University of British Columbia

2008 – 2014: **e-fellows.net** scholarship



## LIST OF PUBLICATIONS

The following list contains all publications accepted by either conferences or journals as of November 14, 2021.

Bucher, D. B., Kufner, C. L., Schlueter, A., Carell, T., and Zinth, W. 2016. "UV-induced charge transfer states in DNA promote sequence selective self-repair," *Journal of the American Chemical Society* (138:1), pp. 186–190.

Bucher, D. B., Schlueter, A., Carell, T., and Zinth, W. 2014. "Watson–Crick Base Pairing Controls Excited–State Decay in Natural DNA," *Angewandte Chemie International Edition* (53:42), pp. 11,366–11,369.

Schlueter, A., Gust, G., Feuerriegel, S., and Neumann, D. 2017. "Minimization Of Investment Costs For Designing Electricity Distribution Networks," 2017 INFORMS Annual Meeting – Houston, Texas October 22-25.

